REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-01-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
5 Dec 2002	REPRINT	

4. TITLE AND SUBTITLE

Proceedings of the
Quantum Computation for Physical Modeling Workshop 2000,
Old Silver Beach, North Falmouth, MA
October 18-19, 2000

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
61102F

6. AUTHORS
Editors: Jeffrey Yepez and Bruce Boghosian*

5d. PROJECT NUMBER
2304

5e. TASK NUMBER
OT

5f. WORK UNIT NUMBER
A1

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory /VSBYA
29 Randolph Road
Hanscom AFB, MA 01731-3010

8. PERFORMING ORGANIZATION REPORT NUMBER
AFRL-VS-HA-TR-2007-1043

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/VSBYA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for Public Release; distribution unlimited.

13. SUPPLEMENTARY NOTES
Reprinted from *Computer Physics Communications,* **146**, (2002) © 2002 Elsevier Science B.V. *Department of Mathematics, Tufts University, Medford, MA 02155.

14. ABSTRACT

A practical and efficient way to use the power of quantum mechanics (quantum parallelism due to the superposition and entanglement of states) could enormously speed up numerical simulations of interest to computational physicists. The diffusion equation, the nonlinear Burgers equations, the turbulent Navier-Stokes equation, and the many-body Schroedinger equation could all be solved much more rapidly if efficient quantum algorithms could be implemented for their solution. The presentations at the conference that are published here review the progress toward these goals.

15. SUBJECT TERMS

Quantum computer Qubit Quantum lattice gas Quantum simulation Electron gas Decoherence
Spin Full adder Fermions Physical modeling Physical quantum algorithms

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Jeffrey Yepez
UNCL	UNCL	UNCL			19B. TELEPHONE NUMBER *(Include area code)*

ELSEVIER

Computer Physics Communications 146 (2002) 277-344

Computer Physics Communications

www.elsevier.com/locate/cpc

Proceedings of the
Quantum Computation for Physical Modeling Workshop 2000
Old Silver Beach, North Falmouth, MA, USA
October 18-19, 2000

Editors: Jeffrey Yepez, Bruce Boghosian

Contents to volume 146

0010-4655/2002 Published by Elsevier Science B.V.
PII: S0010-4655(02)00575-1

ELSEVIER

Computer Physics Communications 146 (2002) 277–279

Computer Physics Communications

www.elsevier.com/locate/cpc

Quantum computation for physical modeling

Jeffrey Yepez

Air Force Research Laboratory, 29 Randolph Road, Hanscom AFB, MA 01731, USA

Foreword

One of the most famous American physicists of the twentieth century, Richard Feynman, in 1982 was the first to propose using a quantum mechanical computing device to efficiently simulate quantum mechanical many-body dynamics [1–3], a task that is exponentially complex in the number of particles treated and is completely intractable by any classical computing means for large systems of many particles. In the two decades following his work, remarkable progress has been made both theoretically and experimentally in the new field of quantum computation [4,5]. Ironically, however, most of the theoretical progress in quantum computing has developed within the purview of the computer scientist with the principle applications of efficient quantum information processing related to cryptography and secure quantum communication.[1] In an effort return to Feynman's original direction, the Air Force Research Laboratory and the Air Force Office of Scientific Research has established a multidisciplinary basic research theme called *Quantum Computation for Physical Modeling* to explore quantum algorithms to model dynamical physical systems. Our goal is to establish a practical and generic means by which the power of quantum mechanics (that is, quantum parallelism due to the superposition and entanglement of states) can be used to speedup numerical simulations of interest to computational physicists.

Notwithstanding the veritable stampede towards computer science related applications by most researchers in the field of quantum computing, a few maverick physicists have developed some quantum algorithms to model quantum mechanical systems. A starting point for this development was a problem posed by Feynman himself to show that the one dimensional Dirac equation could be modeled by a single-speed particle traveling in a two-dimensional space-time as a sum over zigzag paths of straight line elements [6], with the amplitude of a particular path contributing to the kernel by the number of "collisions" or corners along that zigzag path. This *quantum lattice gas* representation of quantum mechanics is equivalent to the well known path integral representation.[2] A quantum lattice gas accounts for all contributing paths by simultaneously evolving many particles in a unitary fashion. Therefore, instead of summing (or integrating over) paths as individual entities, all contributing paths are effectively simulated in one fell swoop as a combined field quantity. In the end, the collisional interaction between particles in the quantum lattice representation can be described by an effective field theory (the Dirac equation in this particular case) at the large-scale called the *continuum limit*.

Beginning in the mid 1990's, a contemporaneous series of quantum lattice-gas algorithms to model the relativistic Dirac equation, equivalent to Feyn-

E-mail address: jeffrey.yepez@hanscom.af.mil (J. Yepez).
URL address: http://qubit.plh.af.mil.

[1] References to specific publications in these subjects are so ubiquitous in the quantum computing literature that we do not include any here. Comprehensive collections of quantum computing papers have been recently published [4,5].

[2] A solution to Feynman's "quantum lattice gas" problem was published in 1984 by Jacobson and Schulman [7].

0010-4655/02/$ – see front matter
PII: S0010-4655(02)00418-6

man's original algorithm, were published by Succi [8, 9], Bialynicki-Birula [10], and Meyer [11–15]. Furthermore, a series of papers on modeling the nonrelativisitic Schröedinger equation were published by Boghosian and Taylor [16–18] and by Yepez and Boghosian [19], the latter article appearing in this issue. Our present goal in the Quantum Computation for Physical Modeling project is accelerate this algorithmic developmental effort that has occurred over the past decade.

In fact, we hope to go further in the application of this quantum algorithmic method. We have developed new efficient quantum lattice-gas algorithms to model classical dynamical systems [20–23]. Meyer also addresses this subject in his article on physical quantum algorithms contained in this issue [24]. In the past, we have considered quantum algorithms suited to globally phase-coherent quantum computers [25], however our focus is presently on those quantum algorithms suited to implementation on locally phase-coherent quantum computers that are technologically much simpler to experimentally implement [26]. This program to use one quantum mechanical system to model another quantum mechanical system or classical system can perhaps best be described as efficient analog computing, which in this context may be termed *analog quantum computing*.

The principle technological obstacle to globally phase-coherent quantum computation is the problem of the uncontrolled decoherence of the quantum computer's wave function. The quantum computing community at large, following the traditional computer scientist's mindset for correcting bit-flip errors using redundancy, has been investing much theoretical work in attempts to develop generalized methods for quantum error correction of bit-flip and phase-change errors [27–29]. As an expedient alternative to this cumbersome approach, as demonstrated by the advent of several nuclear magnetic resonance quantum computing experiments [30–33], it is possible to avoid bit-flip and phase-change errors altogether: *Keep the individual quantum computing elements small enough so that all computation occurs within a single spin-spin decoherence time where bit-flip and phase-change errors are irrelevant.*

Given this possibility of avoiding errors, it is natural to consider building a large-scale quantum computing system by connecting many small ones together in an array interconnected by nearest-neighbor classical communication channels. We call this type of quantum mechanical device a *type-II quantum computer* [26]. This hybrid architecture, combining classical massively parallelism and quantum parallelism, is suited to modeling dynamical physical systems, such as turbulent Navier–Stokes fluids [21–23,25]. In collaboration with the Nuclear Engineering Department of MIT, we have developed a prototype type-II quantum computer based on spatial nuclear magnetic resonance spectroscopy. We use a gradient magnetic field to distinguish individual layers in a liquid sample so that each layer effectively becomes an individual quantum computing node comprising an ensemble of molecules. The first simulation of a quantum lattice-gas model for the one-dimensional diffusion equation [22] has been carried out on this quantum computer prototype using the atomic spin-state of Carbon-13 and Hydrogen nuclei within a linear array of chloroform molecules [34]. This milestone represents the first physical simulation to date on a quantum computer and is contained in this issue. A subsequent paper presenting an improved version of our type-II quantum computer prototype, that corrects for various implementation errors and uses better quantum control, is also in preparation [35].

The rather rapid proof-of-concept achieved by spatial nuclear magnetic resonance spectroscopy of a type-II quantum computer has led to interest in the subject by the Office of the Secretary of the Air Force and the House Appropriations Committee of the US Congress leading in turn to a strong commitment to the field of quantum computation for physical modeling by the Department of Defense. Our new quantum computing research theme is supported by several directorates of the Air Force Office of Scientific Research with about two dozen university research projects across the country to date. The design and construction of several new type-II quantum computer prototypes are now underway using various technologies including superconducting electronics and quantum optics for example.

We have established a new annual workshop series dedicated to this subject of quantum computation for physical modeling (see our web site at http://qubit.plh.af.mil for more details). The first workshop in this series was held in the fall of 2000 in North Falmouth in Cape Cod, Massachusetts and the fol-

lowing collections of articles contained in this issue were contributed from this workshop. Our goal for this workshop series is to annually publish a collection of such contributed articles, to review our progress, and to provide an open forum for new collaborators to join us in this activity.

References

[1] R.P. Feynman, Simulating physics with computers, Internat. J. Theoret. Phys. 21 (6/7) (1982) 467–488.

[2] R.P. Feynman, Quantum mechanical computers, Optics News 11 (2) (1985) 11–20.

[3] A.J.G. Hey, R.W. Allen (Eds.), The Advanced Book Program, Feynman Lectures on Computation, Addison-Wesley, 1996.

[4] C. Macchiavello, G.M. Palma, Z. Zeilinger, Quantum Computation and Quantum Information Theory, World Scientific, 2000.

[5] H.-K. Lo, S. Popescu, T. Spiller, Introduction to Quantum Computation and Information, World Scientific, 1998.

[6] R.P. Feynman, A.R. Hibbs, Quantum Mechanics and Path Integrals, McGraw-Hill, 1965, Problem 2-6 on p. 34.

[7] T. Jacobson, L.S. Schulman, Quantum stochastics: the passage from a relavistic to a non-relativistic path integral, J. Phys. A: Math. Gen. 17 (1984) 375–383.

[8] S. Succi, R. Benzi, Lattice Boltzmann equation for quantum mechanics, Physica D 69 (1993) 327–332.

[9] S. Succi, Numerical solution of the Schrödinger equation using discrete kinetic theory, Phys. Rev. E 53 (2) (1996) 1969–1975.

[10] I. Bialynicki-Birula, Weyl, Dirac, and Maxwell equations on a lattice as unitary cellular automata, Phys. Rev. D 49 (12) (1994) 6920–6927.

[11] D.A. Meyer, From quantum cellular automata to quantum lattice gas, J. Statist. Phys. 85 (5,6) (1996) 551–574.

[12] D.A. Meyer, On the absense of homogeneous scalar unitary cellular automatoa, Phys. Lett. A 223 (1996) 337–340.

[13] D.A. Meyer, Quantum mechanics of lattice gas automata: One-particle plane waves and potentials, Phys. Rev. E 55 (5) (1997) 5261–5269.

[14] D.A. Meyer, Quantum lattice gases and their invariants, Internat. J. Modern Phys. C 8 (1997) 717–735.

[15] D.A. Meyer, Quantum mechanics of lattice gas automata: boundary conditions and other inhomogeneities, J. Phys. A: Math. Gen. 31 (1998) 2321–2340.

[16] B.M. Boghosian, W. Taylor IV, Quantum lattice gas models for the many-body Schrödinger equation, Internat. J. Modern Phys. C 8 (1997) 705–716.

[17] B.M. Boghosian, W. Taylor IV, Simulating quantum mechanics on a quantum computer, Physica D 120 (1998) 30–42.

[18] B.M. Boghosian, W. Taylor IV, A quantum lattice-gas model for the many-particle Schrödinger equation in d dimensions, Phys. Rev. E 57 (1998) 54–66.

[19] J. Yepez, B. Boghosian, An efficient and accurate quantum lattice-gas model for the many-body Schrödinger wave equation, Comput. Phys. Comm. 146 (2002) 280, this issue.

[20] J. Yepez, Lattice-gas quantum computation, Internat. J. Modern Phys. C 9 (8) (1998) 1587–1596, Proceeding of the 7th International Conference on the Discrete Simulation of Fluids, University of Oxford.

[21] J. Yepez, Quantum computation of fluid dynamics, in: C.P. Williams (Ed.), Quantum Computing and Quantum Communications, Lecture Notes in Comput. Sci., Springer-Verlag, 1999, p. 480, First NASA International Conference, QCQC'98, Palm Springs, CA, February 17–20, 1998, Selected Papers.

[22] J. Yepez, Quantum lattice-gas model for the diffusion equation, Internat. J. Modern Phys. C 12 (9) (2001) 1–19, Presented at the 9th International Conference on Discrete Simulation of Fluid Dynamics, Santa Fe, NM, August 22, 2000.

[23] J. Yepez, Quantum lattice-gas model for the Burgers equation, J. Statist. Phys. (2001), To appear, Presented at the 9th International Conference on Discrete Simulation of Fluid Dynamics, Santa Fe, NM, August 22, 2000.

[24] D.A. Meyer, Physical quantum algorithms, Comput. Phys. Comm. (2001) 1–11, quant-ph/011069.

[25] J. Yepez, A quantum lattice-gas model for computational fluid dynamics, Phys. Rev. E (2001) 046702-1–046702-18, APS E-Print: aps1999Oct22_002.

[26] J. Yepez, Type-ii quantum computers, Internat. J. Modern Phys. C 12 (9) (2001) 1–12, Presented at the Quantum Computing Colloquium, MIT Nuclear Engineering Department, September 27, 1999 and at the Air Force Office of Scientific Research Computational and Applied Mathematics Meeting 2000, Stanford University, June 28, 2000.

[27] A.R. Calderbank, P.W. Shor, Good quantum error correcting codes exist, LANL archive: quant-ph/9512032, 1995.

[28] D.G. Cory, M. Price, W. Mass, E. Knill, R. Laflamme, W.H. Zurek, T.F. Havel, S.S. Somaroo, Experimental quantum error correction, Phys. Rev. Lett. 81 (10) (1998) 2152–2155, quant-ph/9802018.

[29] E. Knill, R. Laflamme, L. Viola, Theory of quantum error correction for general noise, Los Alamos National Laboratory Archive, 1999, quant-ph/9908066.

[30] D.G. Cory, A.F. Fahmy, T.F. Havel, Nuclear magnetic resonance spectroscopy: an experimentally accessible paradigm for quantum computing, in: T. Toffoli, M. Biafore, J. Leao (Eds.), PhysComp 96, Proceedings on the Fourth Workshop on Physics and Computation, New England Complex Systems Institute, 1996, pp. 87–91.

[31] J.A. Jones, M. Mosca, Implementation of a quantum algorithm on a nuclear magnetic resonance quantum computer, J. Chem. Phys. 109 (5) (1998) 1648–1653.

[32] I.L. Chuang, N. Gershenfeld, M. Kubinec, Experimental implementation of fast quantum searching, Phys. Rev. Lett. 80 (15) (1998) 3408–3411.

[33] N. Linden, H. Barjat, R. Freeman, An implementation of the Deutsch–Jozsa algorithm on a three-qubit NMR quantum computer, Chem. Phys. Lett. 296 (1998) 61–67.

[34] M. Pravia, Z. Chen, J. Yepez, D.G. Cory, Towards a NMR implementation of a quantum lattice gas algorithm, Comput. Phys. Comm. 146 (2002) 339, this issue.

[35] M. Pravia, Z. Chen, J. Yepez, D.G. Cory, Experimental demonstration of parallel quantum computation, Phys. Rev. E (2002) 1–11, submitted.

ELSEVIER

Computer Physics Communications 146 (2002) 280–294

Computer Physics Communications

www.elsevier.com/locate/cpc

An efficient and accurate quantum lattice-gas model for the many-body Schrödinger wave equation ☆

Jeffrey Yepez [a,*], Bruce Boghosian [b]

[a] *Air Force Research Laboratory, Hanscom AFB, MA, USA*
[b] *Department of Mathematics, Bromfield-Pearson Hall, Tufts University, Medford, MA 02155, USA*

Received 23 April 2001

Abstract

Presented is quantum lattice-gas model for simulating the time-dependent evolution of a many-body quantum mechanical system of particles governed by the non-relativistic Schrödinger wave equation with an external scalar potential. A variety of computational demonstrations are given where the numerical predictions are compared with exact analytical solutions. In all cases, the model results accurately agree with the analytical predictions and we show that the model's error is second order in the temporal discretization and fourth order in the spatial discretization. The difficult problem of simulating a system of fermionic particles is also treated and a general computational formulation of this problem is given. For pedagogical purposes, the two-particle case is presented and the numerical dispersion of the simulated wave packets is compared with the analytical solutions. © 2002 Published by Elsevier Science B.V.

Keywords: Schrödinger wave equation; Quantum computing; Quantum lattice gas; Quantum mechanics; Computational physics

1. Introduction

Feynman's work regarding quantum mechanical computers used to simulate physical quantum mechanical behavior in a numerically efficient way [1–3] has subsequently led to several series of research papers concerned with a variety of details involving the particular quantum algorithm that best represents the Feynman path integral [4,5]. The quantum algorithmic approach is based on a two-component complex field defined on a discrete spacetime lattice where unitary matrices act locally on the field causing its temporal evolution in discrete time steps. Using such a spatially discrete field makes it possible to computationally represent, in the long wavelength limit of modes in the discrete system, the dynamical time-dependent evolution of a continuous wave function in a manner that is numerically efficient.

In 1994 Bialynicki-Birula presented a general quantum algorithmic approach of this kind for modeling the Weyl, Dirac, and Maxwell equations on a body-centered cubic lattice in three-dimensions [6]. In a series of papers on simulating the one-dimensional Dirac equation [7–9], Meyer presented a quantum algorithm similar to that of Bialynicki-Birula with a va-

☆ This work is supported by the Air Force Office of Scientific Research under the Quantum Computation for Physical Modeling initiative.

* Corresponding author.
E-mail addresses: jeffrey.yepez@hanscom.af.mil (J. Yepez), bruce.boghosian@tufts.edu (B. Boghosian).
URL address: http://qubit.plh.af.mil/ (J. Yepez).

0010-4655/02/$ – see front matter © 2002 Published by Elsevier Science B.V.
PII: S0010-4655(02)00419-8

riety of numerical simulations including the effects of boundary conditions, inhomogeneities, and an external scalar potential. Meyer set the quantum algorithm for the discretized path integral in the context of what is called the *quantum lattice gas method* and his algorithm is equivalent to the one-dimensional version of the Bialynicki-Birula quantum algorithm for the Dirac equation.

Contemporaneously with Meyer, two other series of papers on quantum lattice-gas models of the one-dimensional non-relativistic Schrödinger wave equation were presented by Succi and Benzi [10,11] and by Boghosian and Taylor [12–14]. The approach taken by Succi and Benzi is somewhat more computationally oriented in that they begin with a "kinetic" lattice Boltzmann equation of motion[1] (effectively the one-dimensional Dirac equation in the Majorana representation) and show that the Schrödinger wave equation emerges as the governing equation of motion for the slow mode in the long wavelength "hydrodynamic" limit. That is, Succi and Benzi observed that the "macroscopic scale" Schrödinger wave equation arises from the "mesoscopic scale" Dirac equation in a manner quite analogous to how the macroscopic Navier–Stokes hydrodynamic fluid equation arises from the mesoscopic kinetic Boltzmann equation through the Chapman–Enskog expansion.

Boghosian and Taylor's approach follows more along the lines of Meyer's approach in that their model is developed as a generalization of the classical lattice gas method. A kinetic transport equation, now formulated directly at the "microscopic scale", again leads to the Schrödinger wave equation in the continuum limit. The Boghosian and Taylor quantum lattice gas model focuses on solving the many-body Schrödinger wave equation with an arbitrary scalar potential in an arbitrary number of spatial dimensions. They analytically argue for an exponential numerical speedup arising from simulation in the many-body sector of the full Hilbert space carried out simultaneously using quantum superposition of states. The Boghosian and Taylor version of the quantum algorithm is also cast explicitly for direct implementation of an array of quantum bits [13]. Polley has recently presented an argument for adding both an external scalar and vector potential into a quantum lattice-gas model by analytically demonstrating the discrete model's invariance with respect to a general local gauge transformation [18].

A characteristic feature of all these quantum algorithms, used to model the dynamical behavior of either relativistic or non-relativistic quantum particles, is that the governing wave function is well approximated as one approaches the continuum limit where the grid resolution of the spatial lattice become infinite (the lattice cell size approaches zero). Therefore, from the point-of-view of the modeler, there exists a "microscopic scale" where the unitary dynamical rules are locally applied in a discretized fashion and time advances forward in incremental units in a way that is quite artificial. Yet at the "macroscopic scale", which corresponds to the long wavelength limit of the dynamical modes in the discrete system, there is an emergent effective field theory for a complex amplitude field, continuous and differentiable in both space and time, that exactly obeys the physical quantum mechanical equations of motion. At the small scale (characterized by the lattice cell size) one imagines a system of fermionic particles undergoing local collisions and translation to nearby nodes of the lattice. Each of these fermionic particles occupies a local positional state at a specific lattice node with a certain probability amplitude. All the possible locations of the actual physical quantum particle are effectively modeled by the interfering set of probability amplitudes associated with this system of fermionic particles. That is, all the possible pathways are modeled simultaneously using a kind of kinetic system of locally interacting fermions on the small scale.

Seen as a kinetic system then, we may expect that there exists a *local equilibrium configuration of particles*. We require that this configuration be an eigenket, with unity eigenvalue, of the local unitary collision operator that is uniformly and spatially homogeneously applied to the lattice-based two-component field. Then the dynamical lattice-based system on all lattice nodes undergoes local relaxation towards this equilibrium configuration. However, unlike a classical kinetic system, the global configuration of particles does not relax to a single steady-state equilibrium. Instead, there are many steady-state global equilibrium configurations which effectively are the energy eigenstates of the "macroscopic scale" quantum mechani-

[1] The classical lattice Boltzmann equation was popularized with its application to computational fluid dynamics [15–17].

cal equation of motion modeled by the quantum lattice gas. If the quantum lattice gas system is initialized in any one of the energy eigenstate global configurations, the macroscopic scale configuration of the system will remain fixed in time albeit the microscopic configuration of particles would be continually changing at every unit time step.[2] In the end, the Feynman path integral is efficiently and accurately recast as a kinetic dynamical process computed in parallel efficiently on a spacetime lattice.

In this paper we do not argue that the quantum lattice gas dynamical rules represent a local hidden variable theory of quantum mechanics. Although fundamental arguments can be made to limit the possible form of the local unitary operator in the quantum lattice gas [6,7], these arguments lead to an algorithm suited for implementation on a quantum computer. In our investigation of a suitable local unitary collision operator we have found that one quantum gate in particular, the square-root-of-swap given below in Section 2.3, is especially useful for modeling the Schrödinger wave equation in that the local equilibrium configuration discussed above is an eigenket of this gate and with unity eigenvalue. Therefore, we have selected the square-root-of-swap gate as our basic model quantum gate. As demonstrated in Section 2.6, we find that this quantum gate leads to an overall modeling error that is second order in the temporal discretization and fourth order in the spatial discretization. Another point regarding this particular gate is that when measurements are periodically made of the state of qubits in the system, which destroys quantum superpositions and entanglements in the system, the macroscopic scale behavior of the quantum lattice gas system is governed by the classical diffusion equation [19].

This paper is organized as follows. First, in Section 2 we describe the basic quantum algorithm, in particular, how we encode the wave function, how we use two basic quantum gates applied to the qubits at the nodes of the lattice. We first describe the algorithm using matrices and then we describe an equivalent finite difference formulation. Using the microscopic finite difference equations, we then derive the effective field theory at the macroscopic scale where both the lattice cell size and the update time step approach zero. Diffusive ordering holds, as is typical for lattice gas systems, where small-scale temporal fluctuations in the wave function go as the square of the magnitude of the small-scale spatial fluctuations. To confirm that our derivation is correct and to test the validity of the quantum lattice gas model, we test the time-dependent dispersion of a free Gaussian packet. We also test the system when it is initialized in an energy state, which is a fixed macroscopic scale steady-state configuration. We find that as we halve the lattice cell size (double the spatial resolution) the cumulative error in the model drops by a factor of $2^{5.45} = 43.7$.

Second, in Section 3 we show how an external scalar potential may be modeled in the quantum lattice gas system by inducing a local phase rotation in the qubits at each node of the lattice. The qubits at a lattice node are phase rotated by an amount corresponding to the strength of the spatially dependent external potential at that node. We show how this local phase rotation, a kind of gauge transformation, produces an additional potential energy term in the Schrödinger wave equation. We then test the quantum algorithm against two well-known cases of harmonic oscillation in a parabolic potential well and quantum scattering off of and tunneling through a constant potential energy barrier. In both cases, the model behaves as expected.

Third, in Section 4 we test how well the quantum lattice gas can simulate the simultaneous dispersion of two fermionic particles. In the case of multiple particles, the operational quantum gate sequence to handle the many-body case is identical to the single particle case presented in Section 2. Therefore, the implementation of either situation on a quantum computer would be identical as well, except for state-preparation and measurements. However, since at present no quantum computer exists that can test the algorithm presented here, we are forced to consider the implementation on a classical computer and here the implementation for a many-body problem is much more complex than for the single-body problem. Nevertheless, we present a general formulation of the quantum gates in a second quantized representation where the basic computational operations are creation and annihilation of local particle occupancies. The advantage of such an

[2] Given a finite size lattice used for modeling purposes, each local configuration oscillates in time even when the global configuration is a time-independent energy eigenvalue. However, the amplitude of the oscillation does approach zero as the lattice size becomes infinite.

implementation is that it is straightforward to implement the fundamental creation and annihilation operation is a way that respects the anti-commutation relations for fermionic particles. We demonstrate that the macroscopic scale behavior of the quantum lattice gas agrees with the exact time-dependent solution of the two-body wave equation.

Finally, we conclude this paper with a short description of results and share some lessons we learned after using the quantum lattice gas model extensively. We also point out some future directions that may be taken to expand the usefulness of the model.

2. Quantum algorithm for a single free particle

We describe the quantum lattice-gas algorithm for modeling the Schrödinger wave equation by considering the simplest case of a single free particle in a one-dimensional space. In this simple case, the wave function $\psi(x,t)$ obeys the following partial differential equation in the position representation

$$i\bar{h}\frac{\partial\psi(x,t)}{\partial t} = -\frac{\bar{h}^2}{2m}\frac{\partial^2\psi(x,t)}{\partial x^2}, \tag{1}$$

where $\bar{h}$ is Planck's constant and m is the mass of the quantum particle. Here $\psi(x,t)$ is a continuous probability amplitude field (e.g., a continuous complex field).

2.1. Encoding the wave function

To "program" a quantum computer to simulate (1), it is necessary to first formulate an encoding scheme where a collection of qubits is used to store the value of the wave function. Since the number of qubits in any quantum computer is necessarily a finite number, the wave function will have to be approximated in the usual way by discretizing a physically continuous amplitude field into an artificially discrete and finite set of complex numbers. To do this, let us begin with a one-dimensional spatial lattice with L number of nodes. With each node of the lattice we associate a position basis ket denoted by $|x_l\rangle$, where $0 \geqslant l \geqslant L-1$. The discretized system ket in the position basis is

$$|\psi\rangle = \sum_{l=0}^{L-1} c_l|x_l\rangle, \tag{2}$$

where $c_l = \langle x_l|\psi\rangle$ is a complex number. In other words, the basic approach to model the single particle wave function governed by (1) is to express $|\psi\rangle$ as a sum of all the possible ways the particle can be situated on the lattice with a probability amplitude c_l associated with each possible location $|x_l\rangle$.

In our model, we assign two qubits to each node of the lattice, for a total of $2L$ qubits in the whole quantum computer. The qubits that reside at the lth node of the lattice are denoted by $|q_0^l\rangle$ and $|q_1^l\rangle$ and they are used to encode the coefficient c_l of (2) of the position ket for that node. Each qubit is a two-level quantum system $|q_a^l\rangle = \alpha_a^l|0\rangle + \beta_a^l|1\rangle$, where $|\alpha_a^l|^2 + |\beta_a^l|^2 = 1$ for $a = 0$ or 1 and $0 \geqslant l \geqslant L-1$. We consider each qubit to be a container that may or may not be occupied by the quantum particle. The quantum particle is said to *occupy* the ath local state at position x_l when $\beta_a^l = 1$. Similarly, the ath local state at position x_l is said to be *empty* when $\beta_a^1 = 0$.

To see how the qubit encoding works, we write $|\psi\rangle$ in the number representation. In the number representation, each basis state is expressible as the ket $|n_0^0 n_1^0 n_0^1 n_1^1 n_0^2 n_1^2 \cdots n_0^L n_1^L\rangle$, where $n_a^l = 0$ or 1 for all l and a. The Boolean variables n_a^l are called the *number variables* and they correspond to a binary indexing of the basis states in the number representation. Since we are concerned with modeling the one-particle wave equation, we need consider only a subset of all the basis states where only one of the number variables is 1 and all the rest are 0. This subset of all the basis states is called the *one-particle sector*. There are $2L$ such combinations and we shall label these with the binary encoding formula $|2^{2l+a}\rangle$, for $a = 0, 1$ and $0 \geqslant l \geqslant L-1$. Therefore, the system ket in the number representation can be written as

$$|\psi\rangle = \sum_{l=0}^{L-1}\sum_{a=0}^{1}\xi_{2l+a}|2^{2l+a}\rangle, \tag{3}$$

where each ξ_{2l+a} is a probability amplitude (e.g., complex number).

Now for each position ket $|x_l\rangle$ there are two corresponding basis states in the number representation $|2^{2l}\rangle$ and $|2^{2l+1}\rangle$. There are two *interfering possibilities* for a particle to occupy the lth position on the lattice. Therefore, the occupancy probability of the lth node is computed by first summing the probability amplitudes of these corresponding basis states and then

computing the square of the absolute value thereof. In other words, the coefficient c_l in (2) is set equal to the sum of the on-site coefficients in (3)

$$c_l \equiv \xi_{2l} + \xi_{2l+1}. \tag{4}$$

The definition (4) is an essential part of the quantum lattice-gas model presented in this paper. In the section below, where we analytically predict an effective field theory for our artificially discretized model, we explain why we need to make this assignment. We will find that (4) is needed for the predicted effective field theory to accurately approximate the Schrödinger wave equation in the long-wavelength limit, which is also defined below.

2.2. Formulating a suitable gate sequence

We shall require that the algorithmic scheme be at least second order convergent in space, so that as we double the grid resolution (e.g., double the number of qubits) we in turn reduce the numerical error due to the field discretization by a factor of one-quarter. With this type of convergence characteristic, we are assured that we can simulate a wave function governed by the Schrödinger wave equation (1) to any arbitrary degree of accuracy. After we formulate our algorithmic scheme, we will then a posteriori verify by direct numerical simulation that it is indeed at least a second-order convergent numerical scheme. In fact in Section 2.6 we will find that our numerical scheme is fourth-order convergent with an error that goes as δx^4.

To simulate the quantum behavior of the wave function, we seek to develop a sequence of 2-qubit gate operations that will act on a large collection of qubits in the simplest way. We impose the following four simplifying constraints:

(1) All quantum gate operations are homogeneous and independent of space and time.
(2) Only a single quantum gate is used to evolve the wave function and this gate is applied to each lattice node independently (locality).
(3) To provide communication channels between lattice nodes, only the simplest gate is used (e.g., a swap gate).
(4) Because the final value of the computed wave function depends on summing interfering possibilities according to (4), we shall use the Hadamard gate at the very end of the simulation prior to making a measurement of the wave function so that a single qubit at each node will encode the probability amplitude $c_l = \langle x_l|\psi\rangle$ in (2).

With two qubits per node, there are four on-site basis kets, $|0\rangle \otimes |0\rangle \equiv (1,0,0,0)$, $|0\rangle \otimes |1\rangle \equiv (0,1,0,0)$, $|1\rangle \otimes |0\rangle \equiv (0,0,1,0)$, and $|1\rangle \otimes |1\rangle \equiv (0,0,0,1)$. In the context of a quantum lattice-gas model, the unitary matrix $\widehat{U}$ is called the *local collision operator* and the on-site ket $|\nu\rangle \equiv |0\rangle \otimes |1\rangle + |1\rangle \otimes |0\rangle = (0,1,1,0)$ is called the *number density ket*. To have a well behaved local equilibrium associated with the collision process, the local collision operator must have the number density ket as an eigenvector with unity eigenvalue.

2.3. Matrix representation

The quantum gate that we use to evolve the wave function, which is applied independently on a site-by-site basis, is the square-root-of-swap gate

$$\widehat{U} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} - \frac{\mathrm{i}}{2} & \frac{1}{2} + \frac{\mathrm{i}}{2} & 0 \\ 0 & \frac{1}{2} + \frac{\mathrm{i}}{2} & \frac{1}{2} - \frac{\mathrm{i}}{2} & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \tag{5}$$

The reason for calling this the square-root-of-swap gate is that $\widehat{U}^2$ is the swap gate itself

$$\widehat{U} \cdot \widehat{U} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{6}$$

The two nontrivial eigenvalues of $\widehat{U}$ are $\lambda_1 = 1$ and $\lambda_2 = -\mathrm{i}$, with eigenvectors $|\nu_1\rangle = (0,1,1,0)$ and $|\nu_2\rangle = (0,-1,1,0)$, respectively. Also, since (5) causes mixing only between the single-particle basis kets $|0\rangle \otimes |1\rangle$ and $|1\rangle \otimes |0\rangle$, it conserves particle number. So (5) is an appropriate choice for the local collision operator.

The full collision operator, denoted $\widehat{C}$, which acts on the system ket $|\psi\rangle$ is formed by a L-fold tensor product over the local collision operators $\widehat{U}$ applied homogeneously and independently on each node of the lattice

$$\widehat{C} = \bigotimes_{l=0}^{L-1} \widehat{U}. \tag{7}$$

Let us denote the swap operator by $\hat{\chi}_{\mu,\nu}$, where μ and ν index any two qubits in the system. The streaming operator, denoted $\widehat{S}_1$, causes a global shift to the right of the first qubit on all the lattice nodes. Therefore, $\widehat{S}_1$ can be represented by a sequence of swap operators acting on nearest neighbors

$$\widehat{S}_1 = \prod_{l=0}^{(L-1)/2} \hat{\chi}_{2l,2l+2}. \tag{8}$$

In matrix form, $\hat{\chi}$ is a $2^2 \times 2^2$ permutation matrix

$$\hat{\chi} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \tag{9}$$

The algorithm we use to model the Schrödinger wave equation involves multiple applications of the collision operator interleaved with streaming operations as follows:

$$\big|\psi(t+\tau/2)\big\rangle = \widehat{E}_1^{1/2}\big|\psi(t)\big\rangle, \tag{10}$$

where the square root of the evolution operator is

$$\widehat{E}_1^{1/2} = \widehat{S}_1^{\mathrm{T}}\widehat{C}\widehat{S}_1\widehat{C}. \tag{11}$$

Here $\widehat{S}_1^{\mathrm{T}}$ denotes the transpose of $\widehat{S}_1$ and is the inverse of $\widehat{S}_1$. Application of $\widehat{S}_1^{\mathrm{T}}$ causes a global shift to the left of the first qubit on all the lattice nodes. One full time step of the evolution is

$$\big|\psi(t+\tau)\big\rangle = \widehat{E}_1\big|\psi(t)\big\rangle. \tag{12}$$

We use four applications of the collision operator in $\widehat{E}_1$ because $\widehat{C}^4$ is the identity operation. Note that $\widehat{S}_1$ and $\widehat{C}$ do not commute, otherwise (12) would be a trivial evolution equation.

Note that in (12), our choice of streaming the first qubit was arbitrary. A more balanced algorithmic approach would treat both qubits identically. Therefore, we could alternatively define one full time step as

$$\big|\psi(t+\tau)\big\rangle \equiv \widehat{E}_2\widehat{E}_1\big|\psi(t)\big\rangle, \tag{13}$$

where

$$\widehat{E}_2^{1/2} = \widehat{S}_2^{\mathrm{T}}\widehat{C}\,\widehat{S}_2\widehat{C}, \tag{14}$$

and where the streaming operator $\widehat{S}_2$ causes a global shift to the right of the second qubit on all the lattice nodes. The advantage of using the balanced algorithm (13) is that its error is fourth-order in space whereas for the unbalanced algorithm (12) it is only third-order.

2.4. Finite difference formulation

It is possible to specify the quantum algorithm to model the Schrödinger equation without the use of matrices. Instead we can write down a set of finite difference equations, which are equivalent to (10), but perhaps simpler to comprehend at first glance. To do this, let us introduce a new notation for the $2L$ probabilities amplitudes ξ_{2l+a} in (3). We will denote the two complex numbers per lattice node by $\varphi_0(x_l, t_n)$ and $\varphi_1(x_l, t_n)$. That is, we have L-pairs of complex numbers. Then, the quantum algorithmic operations (14) can be expressed as follows:

$$\begin{aligned} &\text{if } \mathrm{mod}(n,4) = 0, \\ &\quad \varphi_0(x_l, t_n) = A^*\varphi_0(x_l, t_{n-1}) + A\varphi_1(x_l, t_{n-1}), \\ &\quad \varphi_1(x_l, t_n) = A\varphi_0(x_l, t_{n-1}) + A^*\varphi_1(x_l, t_{n-1}), \end{aligned} \tag{15}$$

$$\begin{aligned} &\text{if } \mathrm{mod}(n,4) = 1, \\ &\quad \varphi_0(x_l, t_n) = \varphi_0(x_{l-1}, t_{n-1}), \\ &\quad \varphi_1(x_l, t_n) = \varphi_1(x_l, t_{n-1}), \end{aligned} \tag{16}$$

$$\begin{aligned} &\text{if } \mathrm{mod}(n,4) = 2, \\ &\quad \varphi_0(x_l, t_n) = A^*\varphi_0(x_l, t_{n-1}) + A\varphi_1(x_l, t_{n-1}), \\ &\quad \varphi_1(x_l, t_n) = A\varphi_0(x_l, t_{n-1}) + A^*\varphi_1(x_l, t_{n-1}), \end{aligned} \tag{17}$$

$$\begin{aligned} &\text{and if } \mathrm{mod}(n,4) = 3, \\ &\quad \varphi_0(x_l, t_n) = \varphi_0(x_{l+1}, t_{n-1}), \\ &\quad \varphi_1(x_l, t_n) = \varphi_1(x_l, t_{n-1}), \end{aligned} \tag{18}$$

where $A = \frac{1}{2} + \frac{\mathrm{i}}{2}$. The finite-difference equation pair (15) is equivalent to the local collision operation $\widehat{C}$, as is the pair (18). The equation pairs (16) and (18) are equivalent to the streaming operations $\widehat{S}$ and $\widehat{S}^{\mathrm{T}}$, respectively.[3]

This finite-difference representation of the algorithm is nearly identical to that presented by Boghosian and Taylor in 1997 [14] where the two on-site qubits

[3] Noting that $A + A^* = 1$, this set of finite difference equations can be expressed in a more compact way

$$\varphi_0(x_{l+\epsilon}, t_{n+1}) = \varphi_0(x_l, t_n) + \Omega_0,$$

$$\varphi_1(x_l, t_{n+1}) = \varphi_1(x_l, t_n) + \Omega_1,$$

where $\epsilon = (-1)^n$ and $\Omega_0 = A(\varphi_1 - \varphi_0)$ and $\Omega_1 = -\Omega_0$, which has the standard form of a lattice-gas transport equation.

are simultaneously streamed to the left and right after collision operation

$$\begin{aligned}\varphi_0(x_{l+1},t_n) &= A^*\varphi_0(x_l,t_{n-1}) - A\varphi_1(x_l,t_{n-1}),\\ \varphi_1(x_{l-1},t_n) &= -A\varphi_0(x_l,t_{n-1}) + A^*\varphi_1(x_l,t_{n-1}).\end{aligned} \tag{19}$$

They noted that after four time steps, the total amplitude $\psi(x_l,t_n) = \phi_0(x_l,t_n) + \phi_1(x_l,t_n)$ satisfies a finite-difference equation which approximates the Schrödinger equation in the continuum limit. The two essential differences between the improved algorithm (15) through (18) presented in this paper and the quantum algorithm (19) appearing in [20] is that we have alleviated the problem of the occurrence of two non-interpenetrating lattice–gas systems independently evolving on different checker-board sub-lattices and we have doubled the numerical accuracy. This is a problem that occurs when both on-site qubits are simultaneously streamed because streaming only a single qubit at a time, as was done for the quantum lattice-gas model of the diffusion equation [19], causes interactions between all the qubits at each time step.

2.5. The governing partial differential equation

It is straightforward using a symbolic mathematics program, and tedious by hand, to use the update rules (15) through (18) to algebraically determine the value of φ_0 and φ_1 at a later time. With the initial wave function set at t_0, one complete cycle of the algorithm is completed at t_8 (that is, $t_8 - t_0 \equiv \delta\tau$). With the wave function defined as $\psi(x_l,t_n) \equiv \varphi_0(x_l,t_n) + \varphi_1(x_l,t_n)$, the result after one cycle is[4]

$$\begin{aligned}&\psi(x_l,t_8)\\ &\quad= -\frac{1+\mathrm{i}}{2}\psi(x_l,t_0) + \psi(x_{l+1},t_0) + \psi(x_{l-1},t_0)\\ &\qquad -\frac{1-\mathrm{i}}{4}\big[\psi(x_{l+2},t_0) + \psi(x_{l-2},t_0)\big].\end{aligned} \tag{20}$$

Note that (20) is the simplified form of the finite-difference equation at the macroscopic scale when the system is very close to local equilibrium throughout the course of the evolution as $\varphi_0(x,t) = \varphi_1(x,t) \simeq \frac{1}{2}\psi(x,t)$ for all x. The full finite-difference equation is too long to present here, but is given in Appendix A. This result is a finite-difference equation for the following partial differential equation governing the continuous amplitude field $\psi(x,t)$

$$\frac{\partial\psi(x,t)}{\partial t} + \mathcal{O}(\delta t^2) = \frac{\mathrm{i}}{2}\frac{\delta x^2}{\delta\tau}\frac{\partial^2\psi(x,t)}{\partial x^2} + \mathcal{O}(\delta x^4), \tag{21}$$

which is an approximation of (1) where the diffusion constant is $\bar{h}/m = \delta x^2/\delta\tau$ and where δx is the lattice cell size.

If one adds a phase angle ζ to the off-diagonal components of collision operator (5) to obtain a slightly more general collision operator

$$\widehat{U} = \begin{pmatrix} 1 & 0 & 0 & 0\\ 0 & \frac{1}{2}-\frac{\mathrm{i}}{2} & (\frac{1}{2}+\frac{\mathrm{i}}{2})\mathrm{e}^{\mathrm{i}\zeta} & 0\\ 0 & (\frac{1}{2}+\frac{\mathrm{i}}{2})\mathrm{e}^{-\mathrm{i}\zeta} & \frac{1}{2}-\frac{\mathrm{i}}{2} & 0\\ 0 & 0 & 0 & -1 \end{pmatrix}, \tag{22}$$

then the resulting governing partial differential equation will have its transport coefficient dependent on this phase angle as follows:

$$\begin{aligned}&\frac{\partial\psi(x,t)}{\partial t} + \mathcal{O}(\delta t^2)\\ &\quad= \frac{\mathrm{i}}{2\sec\zeta}\frac{\delta x^2}{\delta\tau}\frac{\partial^2\psi(x,t)}{\partial x^2} + \mathcal{O}(\delta x^4).\end{aligned} \tag{23}$$

This allows us to simulate a quantum system where a particle's mass can be arbitrarily large $m = \sec\zeta$. Note that in this case the error is cubic and is proportional to $\sin\zeta$. So for very large masses, the accuracy of the model is reduced to third-order in space. Note that (23) is is valid effectively field theory at the macroscopic scale when the system is very close to local equilibrium where $\varphi_0(x,t) = \varphi_1(x,t) \simeq \frac{1}{2}\psi(x,t)$ for all x.

2.6. Numerical confirmations

To numerically test that the quantum algorithm (12) is indeed equivalent to the finite-difference equation (20) and to see just how good of an approximation of the single-particle Schrödinger equation it is, we have performed two simulations.

[4] Note that the result (20) is accurate up to fourth order in δx only in the situation where the initial system is in local equilibrium defined by $\varphi_0(x_l,t_n) = \varphi_1(x_l,t_n)$. In the more general situation when the system is not in local equilibrium where $\varphi_0(x_l,t_n) \neq \varphi_1(x_l,t_n)$, the result (20) is accurate only up to third order in δx.

In the first simulation, we test the numerical time evolution of a Gaussian packet

$$\psi(x,0) = \frac{1}{\sigma^{1/2}\pi^{1/4}} e^{-x^2/(2\sigma^2)}, \tag{24}$$

where $\ell \geqslant x \geqslant L$ for a lattice of size $L = 64\ell$ and where the packet width is $\sigma = L/10$ as shown in Fig. 1.

The exact analytical solution of (21) is obtained by computing the Fourier components of the energy basis functions

$$a_0 = \frac{1}{L} \int_{-L/2}^{L/2} \psi(x,0)\,dx, \tag{25}$$

$$a_n = \frac{2}{L} \int_{-L/2}^{L/2} \psi(x,0) \cos\left(2n\pi \frac{x}{L}\right) dx. \tag{26}$$

With $\bar{h} = 1$ and $m = 1$, the energy eigenvalues are

$$E_n = \frac{2n^2\pi^2\delta x^2}{L^2\delta t}, \tag{27}$$

and the time-dependent solution to (21) plotted in Fig. 1 is

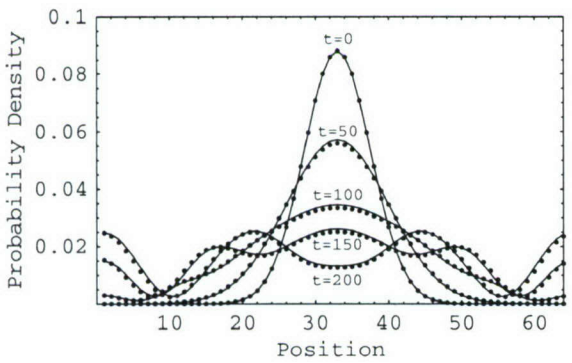

Fig. 1. Time evolution of a Gaussian packet for a single quantum particle overplotted in succession where the x-axis is the position on a 64-node lattice in units of the lattice spacing ℓ and the y-axis is the probability density $|\psi(x,t)|^2$. The solid curves are the exact analytical solution and the circles are the data from the quantum lattice-gas simulation (the initial wave function was normalized, therefore the area under each curve is one). The lattice size is $L = 64\ell$. The initial Gaussian packet of with $\sigma = L/10$ at $t = 0$ is centered at $x = 32\ell$ and the dispersion is evident by observing the wave function at the subsequent times $t = 50\tau, 100\tau, 150\tau$, and 200τ. Periodic boundary conditions were used and $n_{\max} = 20$ energy eigenmodes were used to generate the exact solutions. A time scale factor $t_s = 1.04$ was used to improve the agreement between the numerical and analytical solutions.

$$\psi_{\text{exact}}(x,t) = a_0 + \sum_{n=1}^{n_{\max}} a_n \cos\left(2n\pi \frac{x}{L}\right) e^{-iE_n t/t_s}. \tag{28}$$

Note that in (28), time is scaled by a factor t_s to account for kinetic corrections to the time step. As the number of lattice nodes becomes large, this scaling factor approaches one.

The second test of the quantum lattice-gas algorithm as a model of the Schrödinger wave equation is the measurement of its numerical convergence. Multiple simulations (10 in total) were carried out for lattice sizes ranging from $L = 8\ell, 16\ell, 32\ell, \ldots$ up to $L = 8192\ell$. In each case the initial state of the simulation was the ground state (a sinusoidal energy eigenstate)

$$\psi(x,0) = \psi^{\text{exact}}(x) = \frac{\cos(2\pi x/L)}{\sqrt{L/2}}. \tag{29}$$

Each simulation was run for one time step $T = \tau$ and the numerical error, denoted ϵ, from the exact solution was then measured using the following formula

$$\epsilon(L) = \frac{1}{L} \sqrt{\sum_{x=1}^{L} \left\{ |\psi(x,T)|^2 - |\psi^{\text{exact}}(x)|^2 \right\}^2}. \tag{30}$$

We define the grid resolution as the inverse of the total number of lattice points. That is, for a box of size 1, the *resolving cell size* is defined as $\delta x \equiv \frac{1}{L}$. A plot of the error versus the resolution is given in Fig. 2. As the resolution is increased, the error drops off as $\epsilon(L) \sim L^{-5.45}$.

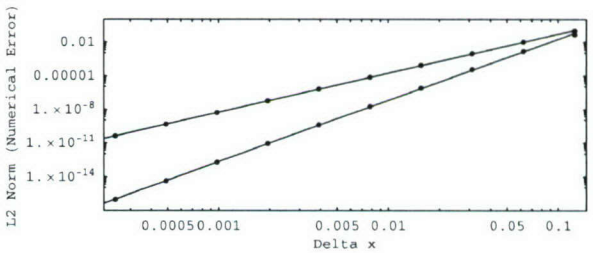

Fig. 2. Log–log plot of the numerical error versus resolving grid cell size, δx, indicating the convergence property of the quantum lattice-gas algorithm (12) and (13) for the Schrödinger equation. The data (black circles) are taken from numerical simulations with grid sizes from $L = 8\ell$ up to 8192ℓ after a single time step $T = \tau$. The solid curves are best-fit linear regression with a slope of 3.48 and 5.45 for the models defined by (12) and (13), respectively. These results demonstrate third-order and fourth-order convergence in space for the two models, respectively.

3. Adding an external scalar potential

It is possible to model an external potential by applying a local phase change to the system wave function [13,14]

$$\psi(x,t) \to \mathrm{e}^{-\mathrm{i}V(x)\delta t}\psi(x,t). \tag{31}$$

The effect of this phase change is to alter the finite difference equation (20) as follows

$$\begin{aligned}\psi(x_l,t_8) = &-\frac{1+\mathrm{i}}{2}\mathrm{e}^{-\mathrm{i}V(x_l)\delta t}\psi(x_l,t_0)\\ &+\left[\mathrm{e}^{-\mathrm{i}V(x_{l+1})\delta t}\psi(x_{l+1},t_0)\right.\\ &\left.+\,\mathrm{e}^{-\mathrm{i}V(x_{l-1})\delta t}\psi(x_{l-1},t_0)\right]\\ &-\frac{1-\mathrm{i}}{4}\left[\mathrm{e}^{-\mathrm{i}V(x_{l+2})\delta t}\psi(x_{l+2},t_0)\right.\\ &\left.+\,\mathrm{e}^{-\mathrm{i}V(x_{l-2})\delta t}\psi(x_{l-2},t_0)\right].\end{aligned} \tag{32}$$

If we expand the potential terms in the arguments of the exponentials

$$V(x_{l+1})\delta t = V(x_l)\delta t + \delta t\delta x\left.\frac{\mathrm{d}V(x)}{\mathrm{d}x}\right|_{x=x_l} + \mathcal{O}(\delta t\delta x^2) \tag{33}$$

we see that we can neglect the second term on the RHS because of diffusive ordering $\delta t\delta x \sim \delta x^3$ since we need to keep terms only to order δx^2. Therefore, in the continuum limit (32) is well approximated by

$$\begin{aligned}&\psi(x_l,t_8)\\ &\quad=-\frac{1+\mathrm{i}}{2}\mathrm{e}^{-\mathrm{i}V(x_l)\delta t}\psi(x_l,t_0)\\ &\qquad+\mathrm{e}^{-\mathrm{i}V(x_l)\delta t}\left[\psi(x_{l+1},t_0)+\psi(x_{l-1},t_0)\right]\\ &\qquad-\frac{1-\mathrm{i}}{4}\mathrm{e}^{-\mathrm{i}V(x_l)\delta t}\left[\psi(x_{l+2},t_0)+\psi(x_{l-2},t_0)\right].\end{aligned} \tag{34}$$

Now multiplying through by $\mathrm{e}^{\mathrm{i}V(x_l)\delta t}$ and expanding the LHS to order δt^2 we have the following finite-difference equation:

$$\begin{aligned}&\left[1+\mathrm{i}V(x_l)\delta t\right]\psi(x_l,t_8)\\ &\quad=\frac{1+\mathrm{i}}{2}\psi(x_l,t_0)\\ &\qquad+\psi(x_{l+1},t_0)+\psi(x_{l-1},t_0)\\ &\qquad-\frac{1-\mathrm{i}}{4}\left[\psi(x_{l+2},t_0)+\psi(x_{l-2},t_0)\right].\end{aligned} \tag{35}$$

In the continuum limit, this finite-difference equation represents the Schrödinger wave equation with an external potential term

$$\begin{aligned}&\frac{\partial\psi(x,t)}{\partial t}+\mathcal{O}(\delta t^2)\\ &\quad=\frac{\mathrm{i}}{2}\frac{\delta x^2}{\delta\tau}\frac{\partial^2\psi(x,t)}{\partial x^2}-\mathrm{i}V(x)\psi(x,t)+\mathcal{O}(\delta x^4).\end{aligned} \tag{36}$$

To confirm the validity of (36) we perform the following numerical simulations that yield results that can be checked against analytical predictions:

(1) Harmonic oscillation of a displaced Gaussian wave packet in a parabolic potential.
(2) Quantum tunneling through a potential barrier.

3.1. Harmonic oscillator

The first numerical test presented here is the simulation of the behavior of a wave packet in an external parabolic potential. This is the well-known problem of the linear harmonic oscillator. Schrödinger analytically calculated the exact time-dependent solution for the evolution of a Gaussian packet that is displaced by a distance a from its central ground state in a parabolic potential well of the form $V(x) = \frac{1}{2}Kx^2$. The initial wave function is

$$\psi(x,0) = \frac{\alpha^{1/2}}{\pi^{1/2}}\mathrm{e}^{-\alpha^2(x-a)^2/2}, \tag{37}$$

where $\alpha = (mK/\bar{h}^2)^{1/4}$ is the width of the packet and $\omega_c = (K/m)^{1/2}$ is the angular frequency of the classical harmonic oscillator [21]. The exact time-dependent solution for the probability density is the following:

$$\left|\psi(x,t)\right|^2 = \frac{\alpha}{\pi^{1/2}}\mathrm{e}^{-\alpha^2(x-a\cos\omega_c t)^2/2}. \tag{38}$$

A derivation of the result (38) is also presented by Schiff [22].

To test the quantum lattice gas algorithm against (38) we used a periodic lattice with $L = 256\ell$ nodes. The initial Gaussian packet is displaced to the right of the center of the grid by 32 lattice nodes and so is initially located at $x_0 = 160\ell$ as shown in Fig. 3. With $\bar{h} = 1$ and $m = 1$, the classical time period is $T_c = 2\pi/\omega_c = 1987\tau$. So letting the simulation run for 1000 iterations allows the packet to the other side

of the potential well near position $x = 96\ell$ as demonstrated in Fig. 3.

The simulation was run for a total of 6000 time steps and the location of the peak of the Gaussian wave packet was recorded every 100τ time steps. This data is plotted in Fig. 4. The location of the peak oscillates in time as expected. Overplotted on this numerical data is the exact solution for the oscillation $a\cos\omega_c t + x_0$

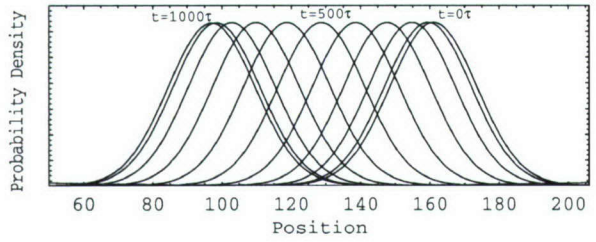

Fig. 3. Time evolution of a Gaussian packet initially displaced by $a = 32\ell$ lattice sites from the center of a parabolic potential well with $K = 10^{-5}$. The width of the packet is $\alpha = 14.4\ell$. The time development of the Gaussian packets over plotted in succession where the x-axis is the position on a $L = 256\ell$ node lattice and the y-axis is the probability density $|\psi(x,t)|^2$. The red curve is the parabolic potential. The $\hbar = 1$ and $m = 1$, the time period of the oscillation is $T_c = \frac{2\pi}{\omega_c} = 1986.92\tau$. A total of ten profiles are over plotted corresponding to time $t = 0, 100\tau, 200\tau, \ldots, 1000\tau$, which is approximately half of the oscillation time period, so the packet is seen to "swing" to the other side of the potential well while maintaining a fixed shape as analytically predicted.

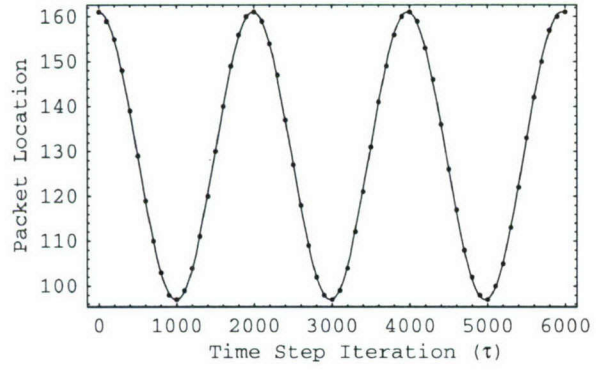

Fig. 4. A comparison between the analytical and numerical predictions of the location of an oscillating Gaussian packet in a harmonic parabolic potential well. The solid curve is the analytical prediction and the black circles are the numerical data taken from the quantum lattice gas simulation presented in Fig. 3. In the simulation, the packet is initially displaced 32 lattice units from the center of the grid at lattice node 128 for a periodic system with a total of $L = 256\ell$ nodes. The numerical predictions are in excellent agreement with the exact analytical solution.

and the agreement between the analytical solution and the numerical data is excellent.

3.2. Scattering off a potential barrier

The next numerical test of the quantum lattice gas is to simulate the well-known case of quantum tunneling through a constant potential barrier of width a. That is, $V(x) = V_0$ for $0 \leqslant x \leqslant a$ and $V(x) = 0$ otherwise. The initial wave function is a Gaussian packet with net momentum to the right

$$\psi(x,0) = \frac{1}{\pi^{1/4}\sigma^{1/4}} e^{-\frac{1}{2}(\frac{x-x_0}{\sigma})^2 + ipx}, \tag{39}$$

where p is the momentum parameter. We choose the mean kinetic energy of the packet to be equal to the constant energy level of the potential barrier $\frac{1}{2}p^2 = V_0$. In this case, the packet tunnels through the barrier but the sum of the transmission and reflection probabilities are less than one because there is a resonance

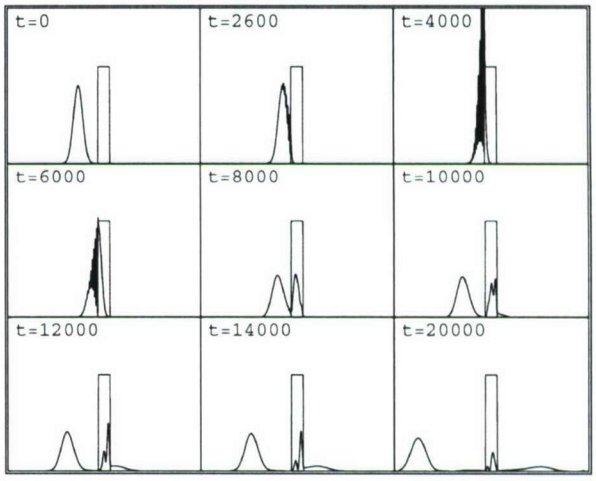

Fig. 5. A sequence of snap shots of the time evolution of a packet that is incident from the left on to a potential barrier where the mean kinetic energy of the packet equals the energy of the barrier. The x-axis is the lattice position and the y-axis is the probability density. The iteration time step for each frame of the sequence is labeled in the upper left corners. The simulation was run on a periodic grid of size $L = 4000\ell$ for a total of 20,000 time steps. The width of the incident packet was set to $\sigma = 0.035L = 140\ell$ and the initial momentum parameter was set to $p = 0.1$ in units where $\ell = 1$, $\tau = 1$ and $m = 1$. The width of the barrier was set to $a = 0.064L = 256\ell$. As expected the numerical simulation clearly demonstrates the resonance effect where there is a non-zero probability of the particle to be trapped within the barrier itself.

effect where the particle is also trapped inside of the barrier. This effect is observed in the numerical simulation shown in Fig. 5.

4. Two fermionic particles

The efficiency of the quantum algorithm (12) becomes evident when it is used to simulate the dynamics of multiple quantum particles. The case of multiple quantum particles is still handled by the same evolution operator $\widehat{E}$ that we tested for the single particle case. The particular sequence and number of quantum gate operations remains fixed, independent of the number of particles to be simulated. The only difference is how the system wave function is initialized.

In this section, for pedagogical reasons, we will consider the case of simulating two free quantum particles. The approach we use in this case can be directly generalized to the many-particle case.

To begin with we write the Schrödinger wave equation for two free quantum particles

$$\mathrm{i}\bar{h}\frac{\partial\psi(x,y,t)}{\partial t} = -\frac{\bar{h}^2}{2m}\frac{\partial^2\psi(x,y,t)}{\partial x^2} - \frac{\bar{h}^2}{2m}\frac{\partial^2\psi(x,y,t)}{\partial y^2}, \tag{40}$$

where x and y are the spatial coordinates of the first and second particle, respectively. Since the wave function is spatially separable as $\psi(x,y,t) = \varphi(x,t)\cdot\varphi(y,t)$, the analytical solution to (40) is obtained in a similar fashion to the one-body case by computing the Fourier components of the energy basis functions

$$a_0 = \frac{1}{L}\int_{-L/2}^{L/2}\varphi(x,0)\,\mathrm{d}x, \tag{41}$$

$$a_n = \frac{2}{L}\int_{-L/2}^{L/2}\varphi(x,0)\cos\left(2n\pi\frac{x}{L}\right)\mathrm{d}x, \tag{42}$$

$$b_n = \frac{2}{L}\int_{-L/2}^{L/2}\varphi(x,0)\sin\left(2n\pi\frac{x}{L}\right)\mathrm{d}x. \tag{43}$$

The energy eigenvalues are still given by (27) and the time-dependent single-particle solution is

$$\varphi(x,t) = a_0 + \sum_{n=1}^{n_{\max}}\left[a_n\cos\left(2n\pi\frac{x}{L}\right) + b_n\sin\left(2n\pi\frac{x}{L}\right)\right]\mathrm{e}^{-\mathrm{i}E_n t/t_s}, \tag{44}$$

which is basically the same as (28) except that we had to add the sin term because with two particles the wavefunction is not even, as is (24), for example. We shall test the time evolution of two Gaussian packets. The initial wave function in our test is the odd function

$$\psi_{\mathrm{exact}}(x,y,t) = \frac{1}{\sqrt{2}}\left[\varphi_{\alpha_1,\sigma_1}(x,t)\varphi_{\alpha_2,\sigma_2}(y,t) - \varphi_{\alpha_1,\sigma_1}(y,t)\varphi_{\alpha_2,\sigma_2}(x,t)\right], \tag{45}$$

where

$$\varphi_{\alpha,\sigma}(x,0) = \frac{1}{\sigma^{1/2}\pi^{1/4}}\mathrm{e}^{-(x-\alpha)^2/(2\sigma^2)}. \tag{46}$$

The subscripts on the function $\varphi_{\alpha,\sigma}$ denote its dependence on the position and width of the individual Gaussian packet. This functional dependence is actually contained within the form of the coefficients a_0, a_n, and b_n that depend on the position and width of the Gaussian packet in accordance with (41) through (43). Note that given the form of (45), $\psi_{\mathrm{exact}}(x,x,t) = 0$ which satisfies the Pauli exclusion principle.

4.1. Numerical confirmation

To numerically simulate the evolution of the two-particle wave function governed by (40) using quantum algorithm (12) we must use a new computational formulation to implement our algorithm. The finite-difference equation implementation that we used in Section 2.4, in the single-particle case, cannot be directly applied in the two-particle case to each particle individually because it does not allow for the possibility when the particles are quantum mechanically entangled. In general, this will be the case when there is an interaction between the particles. Therefore, we shall use an implementation that can handle the most general situations involving correlated particles and one that naturally scales to handle an arbitrarily large number of particles in the system.

We shall represent the basic quantum gate operations in terms of the fermionic creation and annihilation operators in the number representation, denoted $\hat{a}_\alpha^\dagger$ and $\hat{a}_\alpha$, respectively, and use this approach as the basis for a general computational formulation applicable, in particular, to our algorithm and, in general, to any quantum algorithm. Acting on a system of Q qubits, $\hat{a}_\alpha^\dagger$ and $\hat{a}_\alpha$ create and destroy a particle occupancy encoded in the αth qubit

$$\hat{a}_\alpha^\dagger|n_1\ldots n_\alpha\ldots n_Q\rangle = \begin{cases} 0 & n_\alpha = 1, \\ |n_1\ldots 1\ldots n_Q\rangle & n_\alpha = 0, \end{cases} \tag{47}$$

$$\hat{a}_\alpha|n_1\ldots n_\alpha\ldots n_Q\rangle = \begin{cases} |n_1\ldots 0\ldots n_Q\rangle & n_\alpha = 1, \\ 0 & n_\alpha = 0. \end{cases} \tag{48}$$

The fermionic creation and annihilation operators satisfy the anti-commutation relations

$$\begin{aligned} \{\hat{a}_\alpha, \hat{a}_\beta^\dagger\} &= \delta_{\alpha\beta}, \\ \{\hat{a}_\alpha, \hat{a}_\beta\} &= 0, \\ \{\hat{a}_\alpha^\dagger, \hat{a}_\beta^\dagger\} &= 0. \end{aligned} \tag{49}$$

The number operator $\hat{n}_\alpha \equiv \hat{a}_\alpha^\dagger\hat{a}_\alpha$ has eigenvalues of 1 or 0 in the number representation when acting on a pure state, corresponding to the αth qubit being in state $|1\rangle$ or $|0\rangle$, respectively.

The square-root-of-swap gate (5) acting on the on-site qubits indexed by α and $\alpha+1$ can be expressed in terms of the creation and annihilation operators as

$$\begin{aligned} \widehat{U}_{\alpha,\alpha+1} = {} & A^*\hat{n}_\alpha(\mathbf{1}-\hat{n}_{\alpha+1}) - A\hat{a}_\alpha^\dagger\hat{a}_{\alpha+1} - A\hat{a}_{\alpha+1}^\dagger\hat{a}_\alpha \\ & + A^*(\mathbf{1}-\hat{n}_\alpha)\hat{n}_{\alpha+1} + \mathbf{1} - \hat{n}_\alpha - \hat{n}_{\alpha+1}, \end{aligned} \tag{50}$$

where $A = \frac{1}{2} + \frac{\mathrm{i}}{2}$. Also, the swap operator (9) acting between the first qubits indexed by α and β at neighboring nodes can be expressed in terms of the creation and annihilation operators as

$$\hat{\chi}_{\alpha,\beta} = \mathbf{1} - \hat{a}_\alpha^\dagger\hat{a}_\beta - \hat{a}_\beta^\dagger\hat{a}_\alpha - \hat{n}_\alpha - \hat{n}_\beta. \tag{51}$$

The quantum gates (50) and (51) are used to implement the quantum lattice gas collision and streaming operations, respectively [23].

The basis state in the two-particle sector can be labeled with the binary encoding formula $|2^{\alpha-1} + 2^{\beta-1}\rangle$ where the integers α and β are in the ranges $1 \leqslant \alpha \leqslant Q$ and $\alpha+1 \leqslant \beta \leqslant Q$. The number of basis states in this case is the binomial coefficient $\binom{Q}{2}$. The system ket can then be expressed in the two-particle sector as

$$|\psi\rangle = \sum_{\alpha=1}^{Q}\sum_{\beta=\alpha+1}^{Q} \xi_{\alpha,\beta}|2^{\alpha-1} + 2^{\beta-1}\rangle. \tag{52}$$

Since there are two qubits per site, we initialize the wave function using (45) as follows:

$$\begin{aligned} \xi_{\alpha,\beta} \equiv \psi_{\text{exact}}\Bigg(& \left\lfloor \frac{\alpha+1}{2} \right\rfloor - \frac{L+1}{2}, \\ & \left\lfloor \frac{\beta+1}{2} \right\rfloor - \frac{L+1}{2}, 0\Bigg), \end{aligned} \tag{53}$$

where the notation $\lfloor x \rfloor$ means the floor of x and where $Q \equiv 2L$. The floor operation is used so that the initial value of the wave function at each node is divided evenly between each pair of on-site qubits. This is on account of definition (4) that allowed us to have interfering possibilities for a single particle to occupy a single position on the lattice. Moreover in the two particle case, still only a single particle can occupy a single position because of the form of the wave function (45) which is consistent with the anti-commutator relations (49). However, particle one can interfere on-site with itself or with particle two, or vice versa since the particles are indistinguishable.

At this point we have described how we implement the two quantum gates used in our algorithm, how we enumerate the basis states, and how we initialize the two-body wave function in this basis. The only remaining issue left to describe is how we project the two-coordinate wave function $\psi(x, y, t)$ on to a single-coordinate wave function $\psi(x, t)$ that can be plotted on a single physical axis. Because of the underlying lattice in our system, this is straightforward to do by summing out one of the coordinates as follows:

$$\psi(x_l, t_n) \equiv \sum_{m=0}^{L-1} \psi(x_l, y_m, t_n). \tag{54}$$

If $\psi(x_l, y_m, t_n)$ is normalized then so is $\psi(x_l, t_n)$ according to (54). A comparison of the time evolution of the analytical solution (45) and the numerical solution (54) for a lattice with 30 nodes is shown in

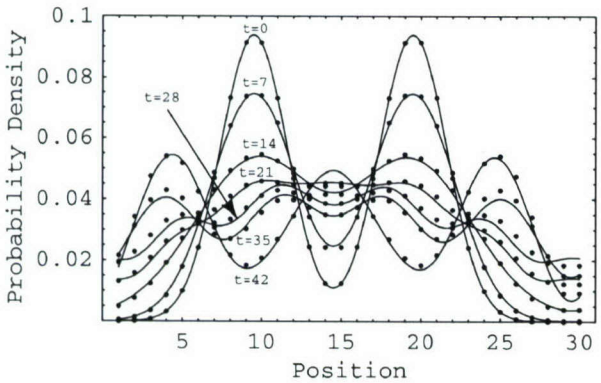

Fig. 6. Time evolution of two fermionic particles initialized as Gaussian packets overplotted in succession where the x-axis is the position on a 30-node lattice in units of the lattice spacing ℓ and the y-axis is the probability density $|\psi(x_1, x_2, t)|^2$ projected onto the x_1-axis. The solid curves are the exact analytical solution and the circles are the data from the quantum lattice-gas simulation (the initial wave function was normalized, therefore the area under each curve is one). The initial Gaussian packets of width $\sigma = 3\ell$ at $t = 0$ of the first and second particle is centered at $x = 10\ell$ and $x = 20\ell$, respectively. The dispersion of both packets is evident by observing the wave function at the subsequent times $t = 7\tau, 21\tau, 28\tau, 35\tau$ and 42τ. Periodic boundary conditions were used and $n_{\max} = 40$ energy eigenmodes were used to generate the exact solutions at four times the resolution of the numerical solution. No time scale factor was used and there is good agreement between the analytical and numerical predictions at all later times of the numerical simulation as demonstated by the graphs.

Fig. 6. Even with this small lattice, throughout the time evolution of the model run the numerical predictions are in good agreement with the predictions of the exact solution.

5. Conclusion

We have presented a quantum algorithm that is an efficient and accurate scheme for simulating the time-dependent evolution of a system of quantum particles governed by the non-relativistic Schrödinger wave equation. The scheme uses a quantum lattice gas system of particles colliding and hopping on a lattice. The algorithm is efficient in the sense that the computational effort needed to simulate an arbitrarily large number of particles (within the constraint of the grid resolution) exactly equals the computational work needed to simulate a single particle, given that the algorithm is executed on a quantum computer that remains phase-coherent throughout the entire coarse of the simulation. However, a limitation does exist on state preparation (i.e. initialize the quantum computer's memory) and we have not argued here that it is possible to initialize the many-body wave function in an efficient way. For that matter, have we also have not argued that it is possible to measure the final state (reading the quantum computer's memory) of the computed wave function in an efficient way. Nevertheless, the quantum algorithm presented here, which is a way of representing a discretized Feynman path integral, has the useful feature that it is explicit in time where the value of the wave function at location x and time $t + \tau$ depends only on the previous values of the wave function at time t in the immediate vicinity of x. Since the algorithm is unitary and fourth order accurate in space, it is useful even for implementation on a classical computer.

We have carried out a variety of numerical tests proving that the quantum algorithm indeed allows us to faithfully reproduce the correct dynamical behavior of a continuous and differentiable wave function in the presence of an external potential. However, the total probability for finding the quantum particle in the system is not exactly conserved in this quantum lattice gas model. One must approach the continuum limit to achieve a high degree of probability conservation. Since we have demonstrated that the quantum lattice gas model is fourth-order convergent in space, it is always possible to choose a grid resolution that achieves the necessary fixed numerical accuracy required by any application.

We have also described and carried out the numerical simulation of two fermionic particles, which are non-interacting except for a quantum mechanical exchange force arising from the anti-commutation relations. The numerical formalism used to implement the quantum lattice gas algorithm represents the basic quantum gates in terms of quadratic products of creation and annihilation operators in a second quantized representation. This formalism straightforwardly handles an arbitrarily large number of particles. The simulation is carried out in the many-body sector (i.e. a Fock space where the number of particles is fixed) where all the basis states are enumerated by a simple binary encoding formula. In general, the wave function is initialized using a Slater determinant so that the Pauli exclusion principle is satisfied and the wave function is odd. In all the test cases (single free par-

ticle, single particle in an external potential, and two free fermions) the numerical predictions agreed extremely well the analytical predictions and exact solutions.

It is important to realize that it is possible to push the quantum lattice gas model into regimes where the numerically predicted results are absolutely wrong. This occurs when the local configurations are far from local equilibrium. Remember that local equilibrium exists on a lattice node when the qubits at that node have identical phase. Therefore, large gradients in the macroscopic profile of the modeled wave function may cause a large phase difference between the on-site qubits eventually resulting in anomalous large-scale behavior in the model. A good example of this occurs when the momentum of a moving wave packet is too high. In this case, since the real and imaginary parts of the wave function are sinusoidal, if the momentum is so high that the wavelength of the traveling wave is on the order of the lattice cell size ($\lambda = \hbar/p \sim \ell$), then after a few times step iterations of the algorithm, large phase differences in the on-site qubits occur. In this case, the local on-site configuration will not relax toward the correct local equilibrium. Fortunately, the norm of the modeled wave function will deviate from unity in these types of cases. Therefore, it is straightforward to test if the model predictions are non-physical by periodically checking the norm of the wave function.

Interaction potentials between particles can also be modeled using this quantum lattice gas method [14]. We still need to perform simulations of a many-body system with an interaction potential. Also, numerical tests in two and three dimensions should be conducted. In this case, it would be straightforward to test the addition of an external vector potential using the results analytically predicted by Polley [18]. The simulation of the dynamical behavior of the positronium atom would be a reasonable next step for the application of the quantum lattice gas method to quantum mechanical computational physics.

Acknowledgements

J. Yepez would like to thank George and Linda Vahala for their critical inspection of this quantum lattice-gas model and helpful discussions.

Appendix A

The full finite-difference equation for the quantum lattice-gas model presented in this paper is quite long. To simplify this expression, we introduce a *local neighborhood vector* with the following 18 components

$$\begin{aligned}\vec{\eta}(x_l,t) &= \big(\varphi_0(x_l,t),\varphi_1(x_l,t),\\ &\quad \varphi_0(x_{l+1},t),\varphi_1(x_{l+1},t),\varphi_0(x_{l-1},t),\varphi_1(x_{l-1},t),\\ &\quad \varphi_0(x_{l+2},t),\varphi_1(x_{l+2},t),\varphi_0(x_{l-2},t),\varphi_1(x_{l-2},t),\\ &\quad \varphi_0(x_{l+3},t),\varphi_1(x_{l+3},t),\varphi_0(x_{l-3},t),\varphi_1(x_{l-3},t),\\ &\quad \varphi_0(x_{l+4},t),\varphi_1(x_{l+4},t),\varphi_0(x_{l-4},t),\varphi_1(x_{l-4},t)\big).\end{aligned} \tag{A.1}$$

We define the following two *coefficient vectors*

$$\begin{aligned}\vec{\alpha} &\equiv (-3,-3\mathrm{i},3,\mathrm{i},-3,-5\mathrm{i},1,\mathrm{i},3,3\mathrm{i},\\ &\quad -1,\mathrm{i},1,3\mathrm{i},0,0,-1,-\mathrm{i}),\\ \vec{\beta} &\equiv (-3\mathrm{i},-3,-5\mathrm{i},-3,\mathrm{i},3,3\mathrm{i},3,\mathrm{i},1,3\mathrm{i},\\ &\quad 1,\mathrm{i},-1,-\mathrm{i},-1,0,0).\end{aligned} \tag{A.2}$$

The microscopic evolution equation (13), explicitly written out, has the following protocol of operations

$$\begin{aligned}\big|\psi(t_{16})\big\rangle \equiv &\big(\widehat{S}_2^{\mathrm{T}}\widehat{C}\,\widehat{S}_2\widehat{C}\,\widehat{S}_2^{\mathrm{T}}\widehat{C}\,\widehat{S}_2\widehat{C}\big)\\ &\times\big(\widehat{S}_1^{\mathrm{T}}\widehat{C}\,\widehat{S}_1\widehat{C}\,\widehat{S}_1^{\mathrm{T}}\widehat{C}\,\widehat{S}_1\widehat{C}\big)\big|\psi(t_0)\big\rangle.\end{aligned} \tag{A.3}$$

The corresponding full finite-difference equation can be specified by the following dot product of these vectors

$$\varphi_0(x_l,t_{16}) = \frac{\vec{\alpha}\cdot\vec{\eta}(x_l,t_0)}{16}, \tag{A.4}$$

$$\varphi_1(x_l,t_{16}) = \frac{\vec{\beta}\cdot\vec{\eta}(x_l,t_0)}{16}. \tag{A.5}$$

Note that if t_0 is the initial time, then the interval $\tau \equiv t_{16}$ is defined the update time step. The finite-difference equation for $\psi = \varphi_0 + \varphi_1$ is

$$\psi(x_l,t_{16}) = \frac{(\vec{\alpha}+\vec{\beta})\cdot\vec{\eta}(x_l,t_0)}{16} \tag{A.6}$$

and it has a high degree of numerical accuracy as indicated in Fig. 2.

References

[1] R.P. Feynman, Simulating physics with computers, Internat. J. Theoret. Phys. 21 (6/7) (1982) 467–488.

[2] R.P. Feynman, Quantum mechanical computers, Optics News 11 (2) (1985) 11–20.

[3] A.J.G. Hey, R.W. Allen (Eds.), The Advanced Book Program, Feynman Lectures on Computation, Addison-Wesley, 1996.

[4] R.P. Feynman, Space-time approach to non-relativistic quantum mechanics, Rev. Modern Phys. 20 (2) (1948) 367–387.

[5] R.P. Feynman, A.R. Hibbs, Quantum Mechanics and Path Integrals, McGraw-Hill, 1965.

[6] I. Bialynicki-Birula, Weyl, Dirac, and Maxwell equations on a lattice as unitary cellular automata, Phys. Rev. D 49 (12) (1994) 6920–6927.

[7] D.A. Meyer, From quantum cellular automata to quantum lattice gas, J. Statist. Phys. 85 (5,6) (1996) 551–574.

[8] D.A. Meyer, Quantum mechanics of lattice gas automata: One-particle plane waves and potentials, Phys. Rev. E 55 (5) (1997) 5261–5269.

[9] D.A. Meyer, Quantum mechanics of lattice gas automata. II. Boundary conditions and other inhomogeneities, quant-ph/9712052:1–24, 1997.

[10] S. Succi, R. Benzi, Lattice Boltzmann equation for quantum mechanics, Physica D 69 (1993) 327–332.

[11] S. Succi, Numerical solution of the Schrödinger equation using discrete kinetic theory, Phys. Rev. E 53 (2) (1996) 1969–1975.

[12] B.M. Boghosian, W. Taylor IV, Quantum lattice gas models for the many-body Schrödinger equation, Internat. J. Modern Phys. C 8 (1997) 705–716.

[13] B.M. Boghosian, W. Taylor IV, Simulating quantum mechanics on a quantum computer, Physica D 120 (1998) 30–42.

[14] B.M. Boghosian, W. Taylor IV, A quantum lattice-gas model for the many-particle Schrödinger equation in d-dimensions, Phys. Rev. E 57 (1998) 54–66.

[15] S. Succi, R. Benzi, F. Higuera, The lattice Boltzmann equation: A new tool for computational fluid dynamics, Physica D 47 (1991) 219–230.

[16] Y.H. Qian, D. d'Humiéres, P. Lallemand, Lattice BGK models for Navier–Stokes equation, Europhys. Lett. 17 (6BIS) (1992) 479–484.

[17] H. Chen, S. Chen, W.H. Mattaeus, Recovery of the Navier–Stokes equations using a lattice-gas Boltzmann method, Phys. Rev. A 45 (8) (1992) R5339–R5342.

[18] L. Polley, Schrödinger equation as the universal continuum limit of nonrelativistic coherent hopping on a cubic spatial lattice, Los Alamos National Laboratory Electronic Archive, quant-ph/9811048:1–6, 2000.

[19] J. Yepez, Quantum lattice-gas model for the diffusion equation, Internat. J. Modern Phys. C 12 (9) (2001) 1–19, Presented at the 9th International Conference on Discrete Simulation of Fluid Dynamics, Santa Fe, NM, August 22, 2000.

[20] B.M. Boghosian, W. Taylor IV, A quantum lattice gas models for the many-body Schrödinger equation, LANL Archive, quant-ph/9604035, 1996.

[21] E. Schrödinger, Naturwiss 14 (1926) 664.

[22] L.I. Schiff, Quantum Mechanics, 3rd edn., International Series in Pure and Applied Physics, McGraw-Hill, New York, 1968.

[23] J. Yepez, A quantum lattice-gas model for computational fluid dynamics, Phys. Rev. E (2001) 046702-1–046702-18, APS E-Print: aps1999Oct22_002.

ELSEVIER

Computer Physics Communications 146 (2002) 295–301

Computer Physics Communications

www.elsevier.com/locate/cpc

Physical quantum algorithms

David A. Meyer

Project in Geometry and Physics, Department of Mathematics, University of California/San Diego, La Jolla, CA 92093-0112, USA

Received 7 September 2001

Abstract

I review the differences between classical and quantum systems, emphasizing the connection between no-hidden variable theorems and superior computational power of quantum computers. Using quantum lattice gas automata as examples, I describe possibilities for efficient simulation of quantum and classical systems with a quantum computer. I conclude with a list of research directions. © 2002 Published by Elsevier Science B.V.

PACS: 03.67.Lx; 05.10.-a

Keywords: Quantum simulation; Quantum lattice gas automata

1. Introduction

There are two paths towards quantum computing: one is teleological and the other is practical. The *teleological* path—described 35 years ago in the prophesy known as Moore's Law [1]—leads down through smaller and smaller device sizes where quantum effects become wilder and wilder. Eventually, rather than domesticating them for classical computation, experimental physicists and engineers believe they will be able to preserve them for quantum computation. The *practical* path, on the other hand, is paved with the desire to solve specific problems efficiently. In an amusing role-reversal, it is theoretical physicists, computer scientists, and mathematicians who follow this path. The first steps along it were taken 20 years ago by Feynman, who suggested that since quantum systems seem to be very hard to simulate on a classical computer, perhaps they could be simulated more efficiently on a quantum computer [2]. More recently, Deutsch [3], Jozsa [4], Simon [5], Shor [6], Grover [7] and others have noted that a quantum computer could solve *classical* problems as well. In this primarily pedagogical paper I describe some of the steps which have been taken along this practical path, and speculate about some steps further along it.

2. Simulating quantum systems classically

Let me begin by reviewing the reasons quantum systems are believed to be hard to simulate on classical computers. Traditionally these are known as 'no hidden variable theorems'. Each is a statement that no classical model with specified constraints can reproduce quantum mechanical results. Consideration of two of them, the Gleason/Kochen–Specker theorem [8,9] and Bell's theorem [10], reveals both their heuristic power and their weaknesses.

In 1957 Gleason proved that for Hilbert spaces of dimension at least 3, any non-negative measure on

E-mail address: dmeyer@chonji.ucsd.edu (D.A. Meyer).

0010-4655/02/$ – see front matter
PII: S0010-4655(02)00420-4

states which is quantum mechanical (i.e. for any orthogonal basis $\{\hat{e}_i\}$ the measure sums to 1) must derive from a density matrix [8]. In 1967 Kochen and Specker made the contradiction with a classical hidden variable model more explicit [9]: They constructed a finite set of unit vectors in $\mathbb{R}^3$ with the property that every attempt to assign values 0 or 1 to each vector satisfying the condition that in each orthogonal triple two vectors get 1 and the third gets 0 must fail. That is, no classical 'hidden variable' can be assigned to pre-determine which outcome of each of some finite set of complete measurements of the spin-squared of a spin-1 particle will be observed (since the spin eigenvalues are $\{-1, 0, +1\}$, two of the spin-squareds are 1 and one is 0 for any complete measurement). Such a hidden variable would be *non-contextual*, in the sense that its value on each vector would specify the spin-squared observed for that measurement, independently of which complete measurement including it is performed. One can argue, however, that noncontextuality is too strict a condition to place on hidden variables—perhaps the results of measurement should depend on hidden variables inherent in the measuring device, which might differ for each complete measurement [11]. Furthermore, the measurements must be exactly along the vectors Kochen and Specker constructed, but from a computational complexity perspective, infinite precision is suspect [12]—and in fact one can show that without additional assumptions one cannot prove a Kochen–Specker theorem using only finite precision measurements [13].

Both of these weaknesses are absent from Bell's theorem [10]. He proved that the results of local measurements on specific states of pairs of spin-$\frac{1}{2}$ particles, i.e. vectors in $\mathbb{C}^2 \otimes \mathbb{C}^2$, cannot be reproduced by any local, classical hidden variable model. Here 'local' means restricted to individual particles. This result is robust against measurement imprecision, and locality of the hidden variables seems justified on physical grounds—the finite speed of light and the locality of physical interactions. In fact, these are the same grounds upon which we base our models of computation: At each timestep a classical or quantum Turing machine changes only the state of the head and the symbol written on the tape cell where the head is located [3,14]; it does not make non-local changes of all the cells of the tape simultaneously, for example.

The states for which Bell's theorem rules out classical hidden variables are *entangled*, i.e. ones for which the state of multiple particles cannot be described as the product of states for each particle individually. Since this is true for all but a set of measure 0 in the space of all pure states, Bell's theorem and its generalizations (see [15] for a recent survey) indicate that most quantum states cannot even be described by reasonable (in the sense of local) classical models. This is a more subtle problem than simply the large size of the state space, which we consider next.

3. Dynamics

The dimension of the Hilbert space describing the state of a system of multiple particles grows exponentially in the number of particles: 2^n for n spin-$\frac{1}{2}$ particles, for example. This exponential explosion, however, is not enough to preclude classical simulation. Consider a classical, probabilistic lattice gas. On a homogeneous one-dimensional lattice of size n there are 4^n basis states s_i, since each lattice site can be occupied by no more than 1 particle with each of the two possible velocities. A general state s is a convex combination:

$$s = \sum_{i=0}^{4^n-1} p_i s_i \quad \text{with} \quad \sum_{i=0}^{4^n-1} p_i = 1, \quad p_i \geqslant 0.$$

Evolving the whole state, i.e. the probability distribution, is therefore an exponentially difficult problem in the size of the lattice. Nevertheless, such lattice gas models are used regularly (see, e.g., [16]). But one does not evolve the whole probability distribution. Rather, one samples it, by evolving a single s_i to a single s_i' at the next timestep, using some random number generator. Multiple runs sample the final distribution. A quantum lattice gas automaton (QLGA, which I will describe in more detail in §4) is also described at each timestep by a vector in a space with basis $\{|s_i\rangle\}$—where $|\cdot\rangle$ is the standard Dirac notation for vectors in Hilbert space [17]:

$$|\psi\rangle = \sum_{i=0}^{4^n-1} a_i |s_i\rangle \quad \text{with} \quad \sum_{i=0}^{4^n-1} |a_i|^2 = 1, \quad a_i \in \mathbb{C}.$$

Evolving the QLGA state has classical computational complexity comparable to evolving the whole state of

the probabilistic LGA. But in the quantum case, this cannot be reduced by sampling individual histories: each has a complex amplitude so the histories with each given final state interfere.

Thus interference seems to be the phenomenon which makes quantum dynamics hard to simulate classically. In fact, although the multi-particle structure of a system is important, entanglement *per se* seems to be less relevant: In liquid state NMR quantum computing experiments [18], for example, the state is not entangled at any timestep (more precisely, since the system is in a mixed state—a convex combination of pure states—the state is *separable)* [19]. Nevertheless, it seems to be difficult to construct a reasonable local hidden variable model for the dynamics [20], i.e. the dynamics seems difficult to simulate classically. To make this more than heuristic, however, we would need a *dynamical* Gleason/Kochen–Specker/Bell-type theorem which applies even for evolution through a sequence of unentangled states. Perhaps some hint about a way to do this may be found in Laflamme's response to the separability criticism of NMR quantum computation [21].

Of course, some demonstrations of the absence of classical models for quantum dynamics already exist. These are more commonly known as quantum algorithms for oracle problems; since each consists of a sequence of unitary operations, they are dynamical results. Grover's quantum search algorithm, for example, solves the problem of identifying $a \in \{0,1\}^n$ given an oracle which responds to a query $x \in \{0,1\}^n$ by returning δ_{xa}, using only $O(\sqrt{2^n})$ queries [7]. Classically, any algorithm would require $O(2^n)$ queries. For $n > 2$, the state is entangled at every timestep after the first [22]. Possibly more to the point is Bernstein and Vazirani's algorithm which solves the problem of identifying $a \in \{0,1\}^n$ given an oracle which responds to a query $x \in \{0,1\}^n$ by returning $x \cdot a \bmod 2$, using only 1 quantum query [23]. Classically, any algorithm would require $O(n)$ queries. And this quantum algorithm works without creating entanglement at any timestep [24]. These results suggest that while a theorem on the impossibility of efficient classical simulation of quantum dynamics may exist, it will have to count *all* the elementary operations, not just the queries, which will presumably make it more difficult to prove.

4. Quantum simulations

In §§1–3 I have tried to explain the heuristic that classical simulation of quantum systems is difficult, while noting what remains to be proved to make such a claim rigorous. Now let us consider Feynman's proposed solution: simulation with quantum computers [2]. The standard model of quantum computation allows polynomially many local (i.e. acting nontrivially on only 1 or 2 qubits) gate operations [25]. This is a reasonable model since in principle it can be realized by a quantum system with a local Hamiltonian. Feynman's proposal has been verified in this model for quantum systems defined by local Hamiltonians [26–31]. More exotic quantum systems can also be simulated efficiently with a standard quantum computer: Fractional quantum Hall systems, for example, have Hamiltonians which vanish on the physical states; the only nontrivial unitary transformations have global (topological) origin. Nevertheless, Freedman, Kitaev and Wang have shown that such topological quantum field theories can be simulated efficiently with a standard quantum computer [32].

A particularly simple architecture for a quantum computer is a QLGA [33]. Although I'm not aware of a demonstration that classical LGA are capable of universal computation, their similarity to the reversible billiard ball model of Margolus [34] suggests that they may be; since QLGA specialize to deterministic LGA, they would be also. Whether they can efficiently (i.e. with polynomial overhead) simulate quantum gate arrays is, I believe, also an open question. In the other direction, QLGA can be simulated efficiently on a standard quantum computer, but have theoretical and possibly practical advantages: They directly simulate quantum systems and are possibly more easily realized experimentally than arbitrary quantum gate arrays.

The possible configurations for each particle on a one-dimensional lattice L are labeled by pairs $(x,\alpha) \in L \times \{\pm 1\}$, where x is the position and α the velocity. A classical lattice gas evolution rule consists of an advection stage $(x,\alpha) \mapsto (x+\alpha,\alpha)$, followed by a scattering stage. Each particle in a QLGA [33] exists in states which are superpositions of the classical states: $|\psi\rangle = \sum \psi_{x,\alpha}|x,\alpha\rangle$, where $1 = \langle\psi|\psi\rangle = \sum \bar{\psi}_{x,\alpha}\psi_{x,\alpha}$. The evolution rule must be unitary; the most general with the same form as the classical rule is:

$$\sum \psi_{x,\alpha}|x,\alpha\rangle \overset{\text{advect}}{\longmapsto} \sum \psi_{x,\alpha}|x+\alpha,\alpha\rangle$$
$$\overset{\text{scatter}}{\longmapsto} \sum \psi_{x,\alpha}S_{\alpha\alpha'}|x+\alpha,\alpha'\rangle,$$

where the scattering matrix is

$$S = \begin{pmatrix} \cos m & \mathrm{i}\sin m \\ \mathrm{i}\sin m & \cos m \end{pmatrix}.$$

Fig. 1 illustrates this quantum evolution: at $m = 0$ it specializes to the classical deterministic lattice gas rule. The $\Delta x = \Delta t \to 0$ limit of this discrete time evolution is the Dirac equation [33]; the $\Delta x^2 = \Delta t \to 0$ limit is the Schrödinger equation [35].

This QLGA model can be extended to include multiple particles with a unitary two particle scattering rule

$$|x,\alpha,x,-\alpha\rangle \mapsto \mathrm{e}^{\mathrm{i}s}|x,\alpha,x,-\alpha\rangle$$

shown in Fig. 1. With these rules the 1-dimensional QLGA discretizes the quantum field theory described by the $(1+1)$-dimensional massive Thirring model [33,36]. These rules also preserve the symmetry (i.e. bosonic or fermionic) of the wave function under particle exchange [37]. The QLGA rules can be generalized to discretize the multi-particle Schrödinger equation in arbitrary dimensions [35]; it seems more difficult, however, to create QLGA which discretize relativistic evolution in higher dimensions [38].

The fact that the QLGA rules are homogeneous, i.e. the same at each lattice site and at each timestep, suggests that they might be easier to implement than general quantum gate arrays which are not. Possible physical systems in which they might be implemented include crystals—as originally proposed by Feynman [2] and more recently in the context of solid state NMR [39]—or optical lattices [40]. A detailed proposal for physical implementation in such systems could motivate experimental work towards realization of QLGA.

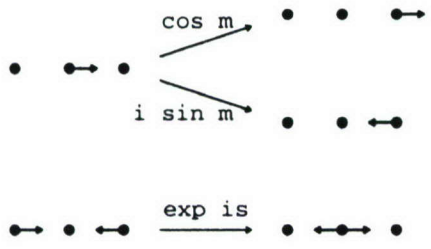

Fig. 1. The general evolution rules for the one-dimensional QLGA.

5. Simulating classical systems

In the previous sections we have seen that there are quantum algorithms to efficiently simulate multiparticle quantum systems which seem to be difficult to simulate classically. Since, as I noted in the Introduction, there are efficient quantum algorithms to solve classical problems, a natural question is whether quantum computers can simulate classical systems [41,42]. Assuming quantum mechanics is a correct description of the world, the existence of a classical description for macroscopic physics means that quantum computers can simulate classical physics with constant overhead—although the constant factor may be something like 10^{23}, i.e. a number of quantum degrees of freedom sufficiently large that subsystems decohere and can be identified as the classical objects to be simulated.

Can we do better? That is, could there be quantum *speedups* for classical physics? Yepez has proposed that the answer is 'yes'. Using a "Type II" quantum computer in which the state is measured, locally, after each timestep and then reset using a lattice Boltzmann rule [43]. A model like this can achieve at most a constant speedup, corresponding to reduced computational cost for local evolution. In practice, of course, a large constant improvement can be tremendously useful, but perhaps it is possible to do better. More precisely, using a standard quantum computer, can classical systems be simulated more efficiently than is possible classically? Lidar and Biham have shown that the answer to this question is also 'yes', for the non-dynamical problem of sampling the ground state distribution of a spin glass [41].

There are also QLGA results which suggest that certain aspects of classical dynamics can be simulated more efficiently quantum mechanically. Consider classical diffusion of a particle in a linear potential, as shown in Fig. 2. A discrete model for the evolution is a biased random walk, with $\mathrm{prob}(\Delta x) \propto \mathrm{e}^{-\nabla V \Delta x}$. The results of simulating the evolution with a classical (probabilistic) LGA are shown in Fig. 3. The average position of the particle satisfies

$$\langle x(t)\rangle - \langle x(0)\rangle \propto -\nabla V t.$$

In QLGA the evolution rules can be modified to include a potential by incorporating an x dependent phase multiplication, i.e. $\mathrm{e}^{-\mathrm{i}V(x)}$ at each timestep [44],

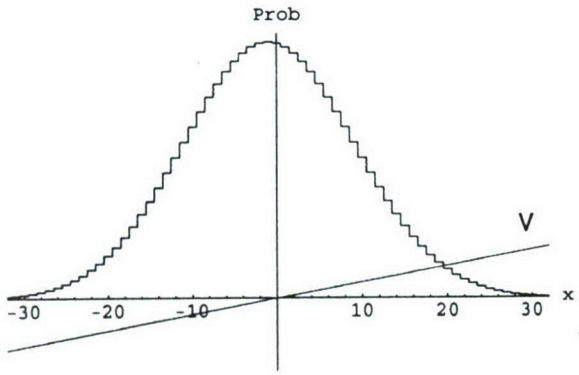

Fig. 2. The probability distribution for the position of a classical particle after diffusing in a linear potential. The particle was initially at the origin.

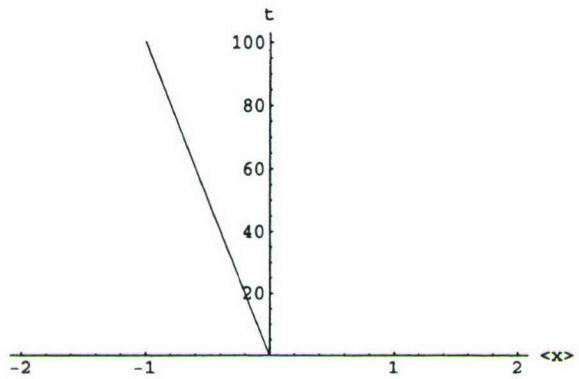

Fig. 3. The evolution of expected position for the random walk model of a classical particle diffusing in a linear potential.

which one might imagine implementing in a physical system with an applied, spatially varying magnetic field, for example. Fig. 4 shows the result of a QLGA simulation with a linear potential. Now the average position of the particle approximately satisfies

$$\langle x(t)\rangle - \langle x(0)\rangle \propto -\nabla V t^2.$$

That is, this quantum system simulates the evolution of the average position of a classical particle diffusing in a linear potential *quadratically faster* than does the classical simulation shown in Fig. 3. I must emphasize that it is only the average position which is being simulated accurately, not the whole probability distribution. Furthermore, the quadratic speedup only holds on timescales $t \ll 2\pi/\nabla V$. On longer timescales the evolution is periodic [45]. Nevertheless, this very simple example suggests that efficient simulation of more complicated classical dynamics may be possible.

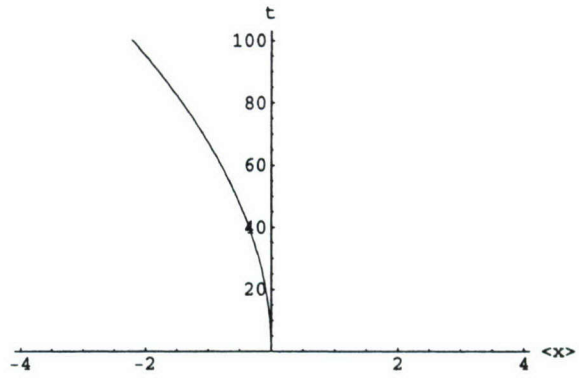

Fig. 4. The evolution of expected position for the QLGA model of a quantum particle subject to a linear potential.

6. Conclusion

In conclusion, let me reiterate the open questions discussed in this paper:

> Is there a proof that (some) quantum dynamics is difficult to simulate classically? Can it be difficult even when the state is unentangled (separable) at each timestep?
>
> In case QLGA become a practical architecture for quantum computers, can they simulate the standard model of quantum computation with no more than polynomial overhead?
>
> What are possible physical implementations of QLGA?
>
> What are the correct QLGA models for relativistic quantum systems in more than 1 spatial dimension?

and most importantly,

> Are there quantum algorithms which speed up the simulation of classical physics?

Positive answers to this last question will broaden the possible uses for a quantum computer and help justify the immense commitment of resources which seems likely to be necessary to develop a scalable one.

Acknowledgements

This work was supported in part by the National Security Agency (NSA) and Advanced Research and Development Activity (ARDA) under Army Research Office (ARO) contract number DAAG55-98-1-0376, and by the Air Force Office of Scientific Research (AFOSR) under grant number F49620-01-1-0494.

References

[1] G.E. Moore, Cramming more components onto integrated circuits, Electronics 38 (1965) 114–117.

[2] R.P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. 21 (1982) 467–488; Quantum mechanical computers, Found. Phys. 16 (1986) 507–531.

[3] D. Deutsch, Quantum theory, the Church–Turing principle and the universal quantum computer, Proc. Roy. Soc. London A 400 (1985) 97–117.

[4] D. Deutsch, R. Jozsa, Rapid solution of problems by quantum computation, Proc. Roy. Soc. London A 439 (1992) 553–558.

[5] D.R. Simon, On the power of quantum computation, in: S. Goldwasser (Ed.), Proceedings of the 35th Symposium on Foundations of Computer Science, Santa Fe, NM, 20–22 November 1994, IEEE Computer Society Press, Los Alamitos, CA, 1994, pp. 116–123; On the power of quantum computation, SIAM J. Comput. 26 (1997) 1474–1483.

[6] P.W. Shor, Algorithms for quantum computation: discrete logarithms and factoring, in: S. Goldwasser (Ed.), Proceedings of the 35th Symposium on Foundations of Computer Science, Santa Fe, NM, 20–22 November 1994, IEEE Computer Society Press, Los Alamitos, CA, 1994, pp. 124–134; Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM J. Comput. 26 (1997) 1484–1509.

[7] L.K. Grover, A fast quantum mechanical algorithm for database search, in: Proceedings of the 28th Annual ACM Symposium on the Theory of Computing, Philadelphia, PA, 22–24 May 1996, ACM, New York, 1996, pp. 212–219; Quantum mechanics helps in searching for a needle in a haystack, Phys. Rev. Lett. 79 (1997) 325–328.

[8] A.M. Gleason, Measures on the closed subspaces of a Hilbert space, J. Math. Mech. 6 (1957) 885–893.

[9] S. Kochen, E.P. Specker, The problem of hidden variables in quantum mechanics, J. Math. Mech. 17 (1967) 59–87.

[10] J.S. Bell, On the Einstein–Podolsky–Rosen paradox, Physics 1 (1964) 195–200.

[11] J.S. Bell, On the problem of hidden variables in quantum mechanics, Rev. Mod. Phys. 38 (1966) 447–452.

[12] A. Schönhage, On the power of random access machines, in: H.A. Maurer (Ed.), Automata, Languages and Programming, Proceedings of the 6th Colloquium, Graz, Austria, 16–20 July 1979, Lecture Notes in Computer Science, Vol. 71, Springer, New York, 1979, pp. 520–529.

[13] D.A. Meyer, Finite precision measurement nullifies the Kochen–Specker theorem, Phys. Rev. Lett. 83 (1999) 3751–3754.

[14] A.M. Turing, On computable numbers, with an application to the Entscheidungsproblem, Proc. London Math. Soc. 442 (1936) 230–265.

[15] R.F. Werner, M.M. Wolf, Quantum Information and Computation 1 (2001) 1–25.

[16] G.D. Doolen, U. Frisch, B. Hasslacher, S. Orszag, S. Wolfram (Eds.), Lattice Gas Methods for Partial Differential Equations, Proceedings, Santa Fe Institute Studies in the Science of Complexity, Vol. IV, Addison-Wesley, New York, 1990.

[17] P.A.M. Dirac, The Principles of Quantum Mechanics, 4th edn., Oxford University Press, Oxford, 1958.

[18] I.L. Chuang, L.M.K. Vandersypen, X. Zhou, D.W. Leung, S. Lloyd, Experimental realization of a quantum algorithm, Nature 393 (1998) 143–146;
D.G. Gory, W. Mass, M. Price, E. Knill, R. Laflamme, W.H. Zurek, T.F. Havel, S.S. Somaroo, Experimental quantum error correction, Phys. Rev. Lett. 81 (1998) 2152–2155;
J.A. Jones, M. Mosca, R.H. Hansen, Implementation of a quantum search algorithm on a nuclear magnetic resonance computer, Nature 393 (1998) 344–346.

[19] S.L. Braunstein, C.M. Caves, R. Jozsa, N. Linden, S. Popescu, R. Schack, Separability of very noisy mixed states and implications for NMR quantum computing, Phys. Rev. Lett. 83 (1999) 1054–1057.

[20] R. Schack, C.M. Caves, Classical model for bulk-ensemble NMR quantum computation, Phys. Rev. A 60 (1999) 4354–4362; subsequent work indicates that there is a local classical model with *exponentially* much local information (N.C. Menicucci, C.M. Caves, Local realistic model for the dynamics of bulk-ensemble NMR information processing, quant-ph/0111152).

[21] R. Laflamme, Review of 'Separability of very noisy mixed states and implications for NMR quantum computing', Quick Reviews in Quantum Computation and Information, http://quickreviews.org/qc/.

[22] D.A. Meyer, N.R. Wallach, Global entanglement in multiparticle systems, quant-ph/0108104.

[23] E. Bernstein, U. Vazirani, Quantum complexity theory, in: Proceedings of the 25th ACM Symposium on Theory of Computing, San Diego, CA, 16–18 May 1993, ACM Press, New York, 1993, pp. 11–20; Quantum complexity theory, SIAM J. Comput. 26 (1997) 1411–1473.

[24] D.A. Meyer, Sophisticated quantum search without entanglement, Phys. Rev. Lett. 85 (2000) 2014–2017.

[25] M.H. Freedman, Foundations of Computational Mathematics 2 (2002) 145–154.

[26] C. Zalka, Efficient simulation of quantum systems by quantum computers, Proc. Roy. Soc. London A 454 (1998) 313–322.

[27] S. Wiesner, Simulations of many-body quantum systems by a quantum computer, quant-ph/9603028.

[28] S. Lloyd, Universal quantum simulators, Science 273 (1996) 1073–1078.

[29] B.M. Boghosian, W. Taylor IV, Simulating quantum mechanics on a quantum computer, Phys. Rev. D 120 (1998) 30–42.

[30] D.S. Abrams, S. Lloyd, Simulation of many-body Fermi systems on a universal quantum computer, Phys. Rev. Lett. 79 (1997) 2586–2589.
[31] G. Ortiz, J.E. Gubernatis, E. Knill, R. Laflamme, Phys. Rev. A 64 (2001) 022319/1-14.
[32] M.H. Freedman, A. Kitaev, Z. Wang, Simulation of topological field theories by quantum computers, quant-ph/0001071.
[33] D.A. Meyer, From quantum cellular automata to quantum lattice gases, J. Statist. Phys. 85 (1996) 551–574.
[34] N. Margolus, Physics-like models of computation, Physica D 10 (1984) 81–95.
[35] B.M. Boghosian, W. Taylor IV, A quantum lattice-gas model for the many-particle Schrödinger equation in d dimensions, Phys. Rev. E 8 (1997) 705–716.
[36] C. Destri, H.J. de Vega, Light-cone lattice approach to fermionic theories in 2D, Nucl. Phys. B 290 (1987) 363–391.
[37] D.A. Meyer, Quantum lattice gases and their invariants, Int. J. Mod. Phys. C 8 (1997) 717–735.
[38] G.V. Riazanov, The Feynman path integral for the Dirac equation, Sov. Phys. JETP 6 (1958) 1107–1113.
[39] F. Yamaguchi, Y. Yamamoto, Crystal lattice quantum computer, Appl. Phys. A 68 (1999) 1–8.
[40] I.H. Deutsch, G.K. Brennen, P.S. Jessen, Quantum computing with neutral atoms in an optical lattice, Fortsch. Phys. 48 (2000) 925–943.
[41] D.A. Lidar, O. Biham, Simulating Ising spin glasses on a quantum computer, Phys. Rev. E 56 (1997) 3661–3681.
[42] J. Yepez, A quantum lattice-gas model for computation of fluid dynamics, Phys. Rev. E 63 (2001) 046702.
[43] J. Yepez, Lattice-gas quantum computation, Int. J. Mod. Phys. C 9 (1998) 1587–1596.
[44] D.A. Meyer, Quantum mechanics of lattice gas automata: One particle plane waves and potentials, Phys. Rev. E 55 (1997) 5261–5269.
[45] D.A. Meyer, H. Blumer, J. Statist. Phys. 107 (2002) 225–239.

ELSEVIER

Computer Physics Communications 146 (2002) 302–316

Computer Physics Communications

www.elsevier.com/locate/cpc

Simulating fermions on a quantum computer ☆

G. Ortiz *, J.E. Gubernatis, E. Knill, R. Laflamme

Los Alamos National Laboratory, Los Alamos, NM 87545, USA

Received 6 February 2001

Abstract

The real-time probabilistic simulation of quantum systems in classical computers is known to be limited by the so-called dynamical sign problem, a problem leading to exponential complexity. In 1981 Richard Feynman raised some provocative questions in connection to the "exact imitation" of such systems using a special device named a "quantum computer". Feynman hesitated about the possibility of imitating fermion systems using such a device. Here we address some of his concerns and, in particular, investigate the simulation of fermionic systems. We show how quantum computers avoid the sign problem in some cases by reducing the complexity from exponential to polynomial. Our demonstration is based upon the use of isomorphisms of algebras. We present specific quantum algorithms that illustrate the main points of our algebraic approach. © 2002 Published by Elsevier Science B.V.

1. Introduction

Because of recent exciting algorithms, like the factoring algorithm of Shor [1] and the search algorithm of Grover [2], that solve difficult problems on a quantum computer using algorithms that would be impractical on a classical computer, it is easy to overlook that the original proposals for quantum computers were for the purpose of solving quantum physics problems [3]. People like Feynman [3] focused on the extent to which such a computer could imitate a specific physical process, suggesting in part that quantum problems were inherently too complex for a classical computer [3].

The obvious difficulty with deterministically solving a quantum many-body problem (of fermions or bosons) on a classical computer is the exponentially large basis set needed (i.e. the dimension of its Hilbert space grows exponentially with the number of degrees of freedom). Exact diagonalization approaches (e.g., the Lanczos method) suffer from this exponential "catastrophe". Viewed the other way around, this basis set scaling is what restricts today's classical computer to simulating only small quantum computers. This point seems indisputable, but should not be taken as proof that quantum systems cannot be simulated on a classical computer. By the same token, the recent claims [4] that quantum computers can simulate all quantum systems efficiently lacks explicit and detailed algorithms for specific problems, and lacks a generic model of quantum computation including the unitary maps (quantum gates) that can be physically implementable. Even if a quantum computer existed, some interesting quantum problems, such as finding the ground state of a general quantum Hamiltonian, do not yet have efficient quantum algorithms. Finding

☆ Work at Los Alamos is sponsored by the US DOE under contract W-7405-ENG-36.

* Corresponding author.
E-mail address: ortiz@viking.lanl.gov (G. Ortiz).

0010-4655/02/$ – see front matter © 2002 Published by Elsevier Science B.V.
PII: S0010-4655(02)00421-6

such a quantity for small systems is relatively routine on a classical computer.

Feynman in fact analyzed two alternatives for simulating physics with computers [3]. One uses a probabilistic classical computer that would produce from the same input as given to a physical system the same distribution of outputs as observed for the physical system. The other uses a computer constructed of distinctively quantum mechanical elements that obey the laws of quantum mechanics. This latter proposal is the quantum computer.

To the question, "Can quantum systems be probabilistically simulated by a classical computer?", Feynman's answer was unequivocally "No". [1] This answer is surprising for even at that time some quantum systems were being very successfully simulated probabilistically on classical computers, mainly by quantum Monte Carlo (QMC) methods [5]. To the question, "Can quantum systems be simulated with a quantum computer?", his answer was a qualified "Yes". He believed almost certainly that this could be done for a system of bosons but was unsure that it could be done for a system of fermions. In this paper we present a design for a universal quantum computer that will simulate a system of fermions. Before doing so, we first discuss some problems that can be solved by a probabilistic simulation of a quantum system on a classical computer and others that cannot.

Probabilistic simulations of quantum systems on a classical computer are mainly performed with the use of the Monte Carlo method. These statistical approaches were introduced to overcome the difficulty of exponentially growing phase spaces by numerically evaluating the accompanying many-dimensional integrals by sampling from a function assumed to be non-negative. On a classical computer one can probabilistically simulate a quantum system like liquid He^4 [6] and produce results that accurately compare with experiment. The situation, however, is far from satisfactory. An unsatisfactory state of affairs results from the frequent breakdown of the non-negativity assumption and is called "the sign problem". The sign problem is manifested by the seemingly exponentially hard task of estimating the expectation value of an observable with a given error. Interestingly, Feynman's negativism about quantum systems being probabilistically simulated by classical computers was a claim that negative probabilities were unavoidable because of the "hidden variable" problem and the possible violation of Bell inequalities. The extent to which the sign problem is a hidden variable problem is unclear. On the other hand, QMC methods do not faithfully adhere to Feynman's idea of a probabilistic computer. Two important differences are that most QMC simulations are non-local and performed in imaginary time. Feynman discussed real-time simulations on a local computer. Implications of these differences have been noted by Ceperley [7] who suggests Feynman really argues just against simulating quantum dynamics on a local classical computer. In any case, the known probabilistic simulations on a classical computer clearly do not qualify as a *universally* efficient computational scheme for general quantum many-body problems. The limiting factors, for whatever reasons, are negative or complex-valued probabilities whether the simulations are done in real or imaginary time.

To place the sign problem in a better perspective, we will start with a real-time analysis of a collection of interacting quantum particles. Quantum mechanics tells us that these particles either obey Bosonic statistics, whereby the wave function is symmetric with respect to the exchange of the states of any two particles, or obey Fermionic statistics, whereby the wave function is antisymmetric (changes sign) with respect to the exchange of any two particles [8]. Examples of bosons are photons and gluons; examples of fermions are electrons, protons, neutrons, and quarks. Often these two quantum statistics conveniently and efficiently map onto a third, quantum spin statistics [9]. Still in other cases, when particle exchange is unlikely, particle statistics is simply ignored.

For a given initial quantum state $|\Psi(0)\rangle$, a quantum computer solves the time-dependent Schrödinger equation

$$i\hbar\frac{\partial|\Psi\rangle}{\partial t} = H|\Psi\rangle \tag{1}$$

by incrementally propagating the initial state via

$$\big|\Psi(t)\big\rangle = \underbrace{e^{-i\Delta t H/\hbar}\cdots e^{-i\Delta t H/\hbar}}_{M\text{ factors}}\big|\Psi(0)\big\rangle. \tag{2}$$

($t = M\Delta t$ and the Hamiltonian H is assumed time independent.) It should be reasonably apparent that if

[1] There is as yet no mathematical proof that this is the correct answer.

the Monte Carlo method is applied to the evaluation of the right-hand side of this equation, it is faced with sampling from oscillatory integrands that are not always positive and have unknown nodal surfaces. Further, as time t increases, the integrand fluctuates with increasing rapidity. While clever stationary-phase forms of the QMC method have been developed, acceptable solutions are possible only for relatively short times. This form of the sign problem is called the *dynamical sign problem*, and we are unaware of any efficient [10] real-time QMC simulations for bosonic, fermionic, or quantum spin systems.

Years ago [5], before quantum computers were proposed, it was realized that by transforming Schrödinger's equation to imaginary-time τ via $t \to -i\hbar\tau$ the problem with the rapid fluctuations was eliminated. With this transformation, called Wick's rotation, one solves the diffusion-like equation

$$\frac{\partial|\Psi\rangle}{\partial\tau} = -H|\Psi\rangle \tag{3}$$

by incrementally propagating the initial state via

$$|\Psi(\tau)\rangle = \underbrace{e^{-\Delta\tau H}\cdots e^{-\Delta\tau H}}_{M \text{ factors}}|\Psi(0)\rangle. \tag{4}$$

($\tau = M\Delta\tau$ and the Hamiltonian H is assumed time independent.) This transformation permits QMC simulations of time-reversal invariant interacting boson systems to a high degree of accuracy. For systems of interacting quantum spins and fermions (or bosons with complex Hermitian Hamiltonians [11]), the transformation does not solve the sign problem. For quantum spin systems, the difficulty is finding a basis in which all matrix elements of the positive-definite operator $\exp(-\Delta\tau H)$ are positive. Most notably this difficulty occurs for frustrated quantum spins. For fermion systems, the problem is the Monte Carlo process causing state exchanges that because of the anti-symmetrization requirement just happen to produce samples which are as frequently positive as negative. For the sign problem found in both types of systems, the statistical error of the measured observables grows exponentially fast with increasing system size. Another difficulty with the imaginary-time approach is analytically continuing the results back to real-time if real-time, i.e. truly dynamical, information is needed [12]. This continuation is an ill-posed problem whose solution places extraordinary demands on the simulation [13].

In this paper, we will focus on the dynamical sign problem for a system of fermions, seemingly the most challenging case. Eventually we will give a detailed implementation of a simulation of the dynamical properties of a collection of interacting fermions on a quantum computer. The implementation avoids the sign problem. First, in Section 2 we will discuss more fully the mathematical origin of the dynamical sign problem in classical computation and show why a quantum algorithm overcomes the problem. In Section 3 we will give the elements required for Deutsch's quantum network model of a quantum computer [14]. The quantum gate in this model conveniently allows the propagation of systems of local two state objects, e.g., a localized quantum spin-$\frac{1}{2}$ particle called qubit. We also propose a universal set of quantum gates (unitary operators) that allows generic propagation of systems of fermions (the fabled "Grassmann chip" [15]). The resulting fermion algebra has been the main technical tool for studying the classical Ising model in two spatial dimensions [16], a prototype lattice model that had an enormous impact on our understanding of phase transitions. Next, in Section 4, we show how this propagation can be effected by the quantum spin gate. We will demonstrate the polynomial scaling of the construction of the initial state, its subsequent time propagation, and the measurement of some observable. Here we will also demonstrate the control of the error in the results. In Section 5, we apply our model of dynamical fermion computation to a toy problem to illustrate our procedures in more detail. Finally, in Section 6, we summarize and make some remarks about future research directions.

Our universal fermion gate and its mapping to the standard universal quantum gate is similar to the one recently discussed by Bravyi and Kitaev [17] who actually propose that a quantum computer built from fermions might be more efficient than one built from distinguishable two state systems.

2. Dynamical sign problem

In order to understand the mathematical origin of the dynamical sign problem we use the Feynman path integral formulation [18] for continuum systems in the first quantization representation. In this formalism one maps a quantum problem in D dimensions into

a classical one in $D+1$ dimensions and then simulates that problem probabilistically on a classical computer. The algorithm is efficient except for the repetition needed to obtain sufficiently good statistics. The "distinguishable particle" quantum mechanical propagator of a system represented by the Hamiltonian $H = \frac{1}{2}\sum_{i=1}^{N_e} p_i^2 + V(\mathcal{R})$ is expressed as [19]

$$G(\mathcal{R} \to \mathcal{R}'; t) = \langle \mathcal{R}', t | \mathrm{e}^{-\mathrm{i}Ht} | \mathcal{R}, 0 \rangle = \int_{\mathcal{R}(0)=\mathcal{R}}^{\mathcal{R}(t)=\mathcal{R}'} \mathcal{D}\big[\mathcal{R}(t)\big] \mathrm{e}^{\mathrm{i}S[\mathcal{R}(t)]}, \tag{5}$$

where the measure

$$\mathcal{D}\big[\mathcal{R}(t)\big] = \lim_{M\to\infty} (2\pi \mathrm{i}t/M)^{-MD/2}\, \mathrm{d}\mathcal{R}_1 \cdots \mathrm{d}\mathcal{R}_{M-1},$$

and the action

$$S\big[\mathcal{R}(t)\big] = \int_0^t \mathrm{d}\tau \left\{ \frac{1}{2}\left(\frac{\mathrm{d}\mathcal{R}(\tau)}{\mathrm{d}\tau}\right)^2 - V\big(\mathcal{R}(\tau)\big)\right\}. \tag{6}$$

Bosonic or fermionic statistics are introduced by applying the corresponding symmetrization operator to the propagator, Eq. (5). However, because the dynamical sign problem occurs for any particle statistics, we will ignore particle statistics for the sake of simplicity.

The description of the properties of different physical systems in terms of correlations of physical observables is the natural way to compare with available experimental information. In this regard, linear response theory provides a way to compute the response of a system to a weak external dynamical perturbation [20]. This linear response is always expressed in terms of a time correlation function of the dynamical variables that couple to the perturbation. For example, if we were to apply an external time-dependent magnetic field and we wanted to calculate the average induced magnetization, we would have to compute a time-dependent magnetization–magnetization correlation function. The two-time correlation function between arbitrary local dynamical variables A and B is

$$C_{AB}(t) = \big\langle A(t)B(0)\big\rangle = \big\langle \mathrm{e}^{\mathrm{i}Ht} A \mathrm{e}^{-\mathrm{i}Ht} B\big\rangle, \tag{7}$$

if the Hamiltonian is time independent. Generically, a stochastic estimate of $C_{AB}(t)$ is

$$C_{AB}(t) = \frac{\sum_{\{\mathcal{R}_i\}} A(\mathcal{R}_{M+1}) B(\mathcal{R}_1) \mathrm{e}^{\mathrm{i}\Phi(\{\mathcal{R}_i\})}}{\sum_{\{\mathcal{R}_i\}} \mathrm{e}^{\mathrm{i}\Phi(\{\mathcal{R}_i\})}} = \frac{\langle A(\mathcal{R}_{M+1}) B(\mathcal{R}_1) \mathrm{e}^{\mathrm{i}\Phi(\{\mathcal{R}_i\})}\rangle_P}{\langle \mathrm{e}^{\mathrm{i}\Phi(\{\mathcal{R}_i\})}\rangle_P}, \tag{8}$$

where the configurations $\{\mathcal{R}_i\}$ are sampled from the probability distribution P (positive semidefinite measure), and Φ is a real-valued function. One immediately sees that the origin of the dynamical sign problem is the oscillatory phase factor $\mathrm{e}^{\mathrm{i}\Phi}$ that leads to large phase fluctuations at long times. Manifestly, $|\langle \mathrm{e}^{\mathrm{i}\Phi(\{\mathcal{R}_i\})}\rangle_P| \to 0$ in an exponential fashion as t gets larger. Therefore, the total statistical error for the evaluation of $C_{AB}(t)$ grows exponentially with time because of large cancellations both in the numerator and denominator. The so-called "fermion sign problem" is a particular case of this problem when $\mathrm{e}^{\mathrm{i}\Phi} = \pm 1$ and time is imaginary [21].

Will a quantum computer solve this problem? One often hears that it will because a quantum computer is a physical system, whether a system of fermions or not, and physical systems have no dynamical or fermion sign problems. Furthermore it has been argued that there are means for mapping most physical systems to a quantum computer in such a way that the quantum computer's controlled evolution mimics that of the physical system [3,22]. A closer look, however, makes the situation less clear. A quantum computer is a computer, and as such it suffers from limited accuracy. More importantly this type of computer predicts results stochastically, meaning each measurement is one member of the ensemble of measurements possible from a distribution specified by the modulus squared of the wave function for the Hamiltonian H modeled by the quantum computer. For a fixed physical time $t > 0$, how accurate is an individual measurement, how accurate is the expectation value of these measurements, and how controlled is their estimated variance? Is the level of accuracy and control achievable polynomially with complexity and t?

There is an area where a problem similar to the sign problem has been recognized and resolved by quantum computation. Recently it was shown that quantum computation is polynomially equivalent to classical probabilistic computation with an oracle for estimating the value of simple sums of rational numbers called *quadratically signed weight enumerators* (QWGTs) [23]. In other words, if these sums could be

evaluated, one could use them to generate the quantum statistics needed to simulate the desired quantum system. More specifically, what was demonstrated was the obtainability of expectation value of operators in quantum computation by evaluating sums of the form

$$S(A, B, x, y) = \sum_{b:Ab=0} (-1)^{b^{\mathrm{T}}Bb} x^{|b|} y^{n-|b|}, \tag{9}$$

where A and B are 0–1-matrices with B of dimension $n \times n$ and A of dimension $m \times n$. The variable b in the summand ranges over 0–1-column vectors of dimension n, b^{T} denotes the transpose of b, $|b|$ is the *weight* of b (the number of ones in the vector b), and all calculations involving A, B and b are done modulo 2. The absolute value of $S(A, B, x, y)$ is bounded by $(|x| + |y|)^n$. Quantum computation corresponds to the problem of determining the sign of $S(A, \mathrm{lt}(A), k, l)$ with the restrictions of having $\mathrm{dg}(A) = I$, k and l being positive integers, and $|S(A, \mathrm{lt}(A), k, l)| \geqslant (k^2 + l^2)^{n/2}/2$. $\mathrm{dg}(A)$ is a diagonal matrix formed from the diagonal elements of A and $\mathrm{lt}(A)$ is a lower triagonal matrix formed from the lower triangular elements of A. Details of this quantum algorithm can be found in [23].

The main point is that these sums are similar to the numerator of Eq. (8), and attempts to estimate them by random sampling result in exponentially bad signal to noise ratios. In the case of QWGTs, quantum computers can estimate the sum exponentially better than classical computers, but the estimate is not exact. The situation for the dynamical sign problem is similar: Quantum computers cannot obtain exact values for the desired correlation functions, but can obtain estimates sufficiently exact to avoid the sign problem suffered by the known classical algorithms and to yield usable information about the physical models simulated.

In this paper we will show explicitly how the sign problem is avoided in the case of simulating fermions. Below we will give a means for translating local fermion Hamiltonians into the Hamiltonians available in the standard model of quantum computation. In contrast to quantum simulations on a classical computer this translation prevents uncontrolled exchange processes that are the dominant source of the fermion sign problem. With respect to the dynamical sign problem, we then argue by using standard error correction analysis developed for the standard model of quantum computing that these gates will enable sufficiently accurate measurements of correlation functions so the accuracy of the average of these measurements will be dominated by the statistical error. The statistical error is problem dependent but polynomially bounded, so that the difficulty associated with phase-weighted averages is eliminated.

3. Models of quantum computation

The *quantum control* model of quantum computation assumes the existence of physical systems that can be controlled by modulating the parameters of the system's Hamiltonian H_P. The control possibilities are abstracted and used to implement specific *quantum gates* that represent the unitary evolution of the physical system over a time step obtained by specific modulations of the Hamiltonian. In most treatments, the physical systems, together with the gates, are then taken as the abstract model of quantum computation. The quantum control and quantum gate viewpoints are effectively equivalent, but to tie the computational model to the physics simulation problem more closely, we choose to describe quantum computation from the point of view of quantum control; that is, we will assume an H_P. In this context we begin by giving the standard model of quantum computation and then defining an alternative model based on fermions.

Defining a model of quantum computation consists of giving an algebra of operators, a set of controllable Hamiltonians (Hermitian operators in the algebra), a set of measurable observables, and an initial state of the physical system. In the simplest case, the observables are measured by the method of projective measurements, and the initial state of the physical system is an expectation value of the algebra's operators.

3.1. Standard model of quantum computation

The standard model of quantum computation (Deutsch's quantum network representation) is based on an assembly of two state systems called *qubits*, controlled by one- and two-qubit Hamiltonians, and on a measurement process determined by one-qubit observables.

Operator algebra. It is convenient to define the standard model through the algebra of operators acting on the qubits. This algebra is generated by the unit and Pauli matrices σ_x, σ_y and σ_z for each qubit j,

$$\mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}; \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}; \\ \sigma_y = \begin{pmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix}; \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{10}$$

These matrices represent quantum operators with mixed commutation relations and span the space of complex-valued 2×2 matrices. For qubits $j \neq k$, the σ's commute, and for qubits $j = k$, they satisfy the relation $\sigma_\mu \sigma_\nu + \sigma_\nu \sigma_\mu = 2\delta_{\mu\nu}\mathbb{1}$ ($\mu, \nu = x, y, z$). For a *quantum register* with n qubits, one may take the operator σ_μ^j defined in terms of a Kronecker product

$$\sigma_\mu^j = \mathbb{1} \otimes \mathbb{1} \otimes \cdots \otimes \underbrace{\sigma_\mu}_{j\text{th factor}} \otimes \cdots \otimes \mathbb{1}$$

of matrices acting on n two-dimensional linear spaces. Thus σ_μ^j admits a matrix representation of dimension $2^n \times 2^n$.

Control Hamiltonians. Control of qubits is achieved by applying Hamiltonians that act on either one or two qubits. A theorem [24,25] in quantum information processing says that a generic operation on a single qubit and any interaction between two qubits is sufficient for building any unitary operation. We take

$$H_P(t) = \sum_j \left[\alpha_{x_j}(t)\sigma_x^j + \alpha_{y_j}(t)\sigma_y^j\right] + \sum_{i,j} \alpha_{ij}(t)\sigma_z^i\sigma_z^j,$$

where the $\alpha_\mu(t)$ are controllable. Ideally, no constraints on the control functions are assumed. However, it is often simpler to design the required control by assuming that only one of the $\alpha_\mu(t)$ is nonzero at any time. A quantum algorithm for this model of quantum computation consists of prescribing the control functions [26]. A convenient measure of the complexity of such an algorithm is the integral $\int_0^t \mathrm{d}t' \sqrt{\sum_\mu \alpha_\mu^2(t')}$ (the *action* of the algorithm). The quantum gates are simply specific unitary evolutions that may be implemented in terms of H_P. A convenient universal set of gates is given by operators of the form $\exp(\mathrm{i}\sigma_\mu^i \pi/4)$ and $\exp(\mathrm{i}\sigma_z^i \sigma_z^j \pi/8)$. In the quantum network representation of the standard model, an algorithm is a specific sequence of these operators applied to the initial state of the qubits.

Initial state. The initial state of the qubits is assumed to be an n term Kronecker product of the state $|0\rangle \equiv \binom{1}{0}$ which is an eigenstate of σ_z with eigenvalue 1. The state is completely determined by the expectation values $\langle 0|\sigma_{\mu_i}^i|0\rangle$, which are 1 if the $\sigma_{\mu_i}^i$ are all σ_z^i or the identity, and are 0 otherwise. Physically, the initial state has all "spins" up.

Measurement. The final feature of the model of computation is the specific means for extracting information after a sequence of operations has been applied to the initial state. In the standard model, it is always possible to apply a projective (von Neumann) measurement [27] using the observables σ_z^i. With this capability, it is unnecessary to give an initial state explicitly, as the desired state can be prepared by using measurement and operations. To learn the expectation of an observable at the end of an algorithm, one repeats the algorithm and measurement procedure many times and averages over the measurements until the desired accuracy is achieved.

For a description of the standard model of quantum computation in terms of quantum Turing machines, see [28]. Quantum networks are discussed in [24]. Introductory descriptions of the standard model may be found in [29,30].

3.2. Fermion model of quantum computation: Grassmann chip

Somewhat analogously, we now describe a standard model of fermion computation. For simplicity we only consider spinless fermions, i.e. fermions without internal spin degrees of freedom, although we could have considered more general fermionic algebras with internal degrees of freedom [9]. Physically, a system of spinless fermion might be a system of spin-$\frac{1}{2}$ electrons in a magnetic field sufficiently strong to polarize it fully. The basic system of this model is a state (or fermionic mode) that can be occupied by 0 or 1 spinless fermion. We define the model for n such modes.

Operator algebra. We define the model through the algebra of the spinless fermion operators a_j and $a_j^\dagger$ for each qubit j ($j = 1, \ldots, n$), i.e. through the algebra of $2n$ elements satisfying canonical anticommutation relations

$$\{a_i, a_j\} = 0, \{a_i, a_j^\dagger\} = \delta_{ij},$$

where $\{A, B\} = AB + BA$ denotes the anticommutator or Jordan product. $a_j^\dagger$ (a_j) creates (annihilates) a spinless fermion in state (mode) j. Each element admits a matrix representation of dimension $2^n \times 2^n$. The fermion algebra is isomorphic (as a $*$-algebra) to the standard model (or Pauli) algebra. The isomorphism is established through the Jordan–Wigner mapping [31].

Control Hamiltonians. We take

$$H_P = \sum_j [\alpha_j(t) a_j + \tilde{\alpha}_j(t) a_j^\dagger] + \sum_{ij} [\alpha_{ij}(t)(a_i^\dagger a_j + a_j^\dagger a_i) + \beta_{ij}(t) a_i^\dagger a_i a_j^\dagger a_j].$$

This is a universal Hamiltonian, i.e. any other Hamiltonian for a system of interacting spinless fermions can be generated by it. Physical operators must be (Hermitian) products of even degree involving combinations of the creation and annihilation operators such as the terms in the last two summands of the Hamiltonian above.

Initial state. The initial state is assumed to be an n term Kronecker product of the state $|0\rangle$ which is an eigenstate of the number operator $a_j^\dagger a_j$ with eigenvalue 0. The state is completely determined by the expectation values $\langle 0|a_j^\dagger a_j|0\rangle = 0$ for all j. Physically, the initial state has all modes unoccupied.

Measurement. Measurements can again be performed by using von Neumann's scheme of projective measurements. In Section 4.3, we will discuss another scheme more appropriate for the physical systems at hand.

In the next subsection we show how to simulate the fermion model by using the standard spin-$\frac{1}{2}$ model. In particular it is possible to efficiently map the fermion Hamiltonians to Pauli operators which can be simulated using the control Hamiltonians of the standard model. This establishes that these two models of computation are polynomially equivalent. Here the point of view is similar to the one used for classical models of computation: the simulation of one model by another establishes their equivalence.

4. Fermion computation via the standard model

In the previous Section we gave the elements required for Deutsch's quantum network model of a quantum computer [14] and proposed a universal set of quantum gates (unitary operators) that allows generic propagation of systems of fermions (the fabled "Grassmann chip" [15]). Here we show how this propagation can be effected by the quantum spin gate. We will demonstrate the polynomial scaling of the construction of the initial state, its subsequent time propagation, and the measurement of some observable. We will also demonstrate the control of the error in the results.

The first step is the observation that the set of $2n$ matrices γ_μ (of dimension $2^n \times 2^n$) satisfying the Clifford algebra identities

$$\{\gamma_\mu, \gamma_\nu\} = 2\delta_{\mu\nu}$$

admits a representation in terms of Pauli matrices (Brauer–Weyl construction)

$$\gamma_1 = \sigma_x^1, \quad \gamma_2 = \sigma_y^1,$$
$$\gamma_3 = \sigma_z^1 \sigma_x^2, \quad \gamma_4 = \sigma_z^1 \sigma_y^2,$$
$$\gamma_5 = \sigma_z^1 \sigma_z^2 \sigma_x^3, \quad \gamma_6 = \sigma_z^1 \sigma_z^2 \sigma_y^3,$$
$$\vdots$$
$$\gamma_{2n-1} = \left[\prod_{j=1}^{n-1} \sigma_z^j\right] \sigma_x^n, \quad \gamma_{2n} = \left[\prod_{j=1}^{n-1} \sigma_z^j\right] \sigma_y^n.$$

The following mapping of fermion operators

$$a_j \to \left(\prod_{i=1}^{j-1} -\sigma_z^i\right) \sigma_-^j = (-1)^{j-1} \sigma_z^1 \sigma_z^2 \cdots \sigma_z^{j-1} \sigma_-^j = (-1)^{j-1} \frac{\gamma_{2j-1} - i\gamma_{2j}}{2},$$

$$a_j^\dagger \to \left(\prod_{i=1}^{j-1} -\sigma_z^i\right) \sigma_+^j = (-1)^{j-1} \sigma_z^1 \sigma_z^2 \cdots \sigma_z^{j-1} \sigma_+^j = (-1)^{j-1} \frac{\gamma_{2j-1} + i\gamma_{2j}}{2},$$

where $\sigma_{\pm}^{j} = (\sigma_x^j \pm \mathrm{i}\sigma_y^j)/2$, defines a $*$-algebra isomorphism to the algebra of operators of the standard model. It is the so-called spin-$\frac{1}{2}$ Jordan–Wigner transformation [31], and has the property that $\hat{n}_j = a_j^\dagger a_j = \sigma_+^j \sigma_-^j = \frac{1}{2}(\mathbb{1} + \sigma_z^j)$. We note that $\hat{n}_j$ is a "local" particle number (or density) operator and many types of interaction in physical systems are of the form "density times density" which simplifies the simulation as we will see.

It is important to emphasize that the success of our approach depends upon the mapping of algebras (and not of Hilbert spaces). In this regard it is relevant to mention that the transformation just presented is a particular case of a more general set of mappings that we would like to name generalized Jordan–Wigner transformations [9]. It is possible to imagine a quantum computer implemented, for example, with a three state unit ($S = 1$) instead of a qubit. In such a case, these generalized transformations still allow one to simulate fermions or particles with arbitrary statistics.

Two additional comments are in order: The mapping for a_j and $a_j^\dagger$ described above corresponds to a one-dimensional array of spins. The extension to higher spatial dimensions can be done [9,32,33] in various ways. A straightforward extension to two dimensions is to re-map the sites of a two-dimensional array onto a one-dimensional string and proceed as before. Also there is nothing special about using the fermion instead of a quantum spin as an alternative model of computation. One could have just as well used the hard core boson [9]. The main question is whether different algebras admit a physical realization. For hard-core bosons this realization is He^4 atoms.

4.1. Evolution

Given a fermion model algorithm, it is necessary to efficiently obtain a corresponding standard model algorithm that at least approximates the desired evolution. The general principle is to map the time-dependent fermion Hamiltonian $H(t) = \sum_i H_i$ to the standard model operators via the Jordan–Wigner transformation, express the result in terms of a sum of simple products of Pauli operators, and then use the Trotter approximation

$$\mathrm{e}^{-\mathrm{i}\Delta t(H_0+H_1+\cdots)/\hbar} = \prod_i \mathrm{e}^{-\mathrm{i}\Delta t H_i/\hbar} + \mathcal{O}\big((\Delta t)^2\big). \quad (11)$$

Each time step Δt is chosen so that the final error of the simulation is sufficiently small. Provided that the number of terms in the sum is polynomially bounded in the number n of qubits or fermionic modes and provided that each term can be polynomially simulated, the simulation is efficient in n and $1/\text{error}$.

To see how to do the simulation, consider the example of the bilinear operator $H_c = a_1 a_j^\dagger + a_j a_1^\dagger$ in the control Hamiltonian of the fermion model:

$$\begin{aligned} H_c &= (-1)^j \big[\sigma_-^1 \sigma_z^1 \cdots \sigma_z^{j-1} \sigma_+^j \\ &\qquad + \sigma_z^1 \sigma_+^1 \sigma_z^2 \cdots \sigma_z^{j-1} \sigma_-^j\big] \\ &= \frac{(-1)^j}{2}\big[\sigma_x^1 \sigma_z^2 \cdots \sigma_z^{j-1} \sigma_x^j + \sigma_y^1 \sigma_z^2 \cdots \sigma_z^{j-1} \sigma_y^j\big]. \end{aligned}$$

It is readily checked that the Jordan–Wigner transformation for the other terms in the control Hamiltonians are also decomposable into sums of a few products of Pauli operators.

The whole idea of a quantum computer is simulating the operations we want by using unitary matrices $U = \exp(-\mathrm{i}\Delta t H_P/\hbar)$. These unitary matrices, representing quantum gates, perform reversible computation and are case-dependent. For our particular case, we know how to simulate $H = \sigma_z^1$ in the spin-$\frac{1}{2}$ case (it is directly implemented in the standard model), so we ask what set of unitary operations produce the evolution $\widetilde{U} = \exp(-\mathrm{i}\Delta t H_c/\hbar)$. In other words, how do we write a $U = U_1 \ldots U_k$ such that $H_c = U^\dagger H U$? Consider for example the Hamiltonian $H_x = \sigma_x^1 \sigma_z^2 \cdots \sigma_z^{j-1} \sigma_x^j$. The procedure is as follows: The unitary operator

$$\begin{aligned} U_1 &= \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_y^1} = \frac{1}{\sqrt{2}}\big[\hat{\mathbb{1}} + \mathrm{i}\sigma_y^1\big] \\ &= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \otimes \mathbb{1} \otimes \cdots \otimes \mathbb{1} \end{aligned}$$

takes $\sigma_z^1 \to \sigma_x^1$, i.e. $U_1^\dagger \sigma_z^1 U_1 = \sigma_x^1$. The operator

$$U_2 = \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_z^1\sigma_z^2} = \frac{1}{\sqrt{2}}\big[\hat{\mathbb{1}} + \mathrm{i}\sigma_z^1\sigma_z^2\big]$$

takes $\sigma_x^1 \to \sigma_y^1 \sigma_z^2$. The next step is

$$U_3 = \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_z^1\sigma_z^3}$$

to take $\sigma_y^1\sigma_z^2 \to -\sigma_x^1\sigma_z^2\sigma_z^3$. By successively similar steps we easily build the required string of operators: $\sigma_x^1\sigma_z^2\cdots\sigma_z^{j-1}\sigma_x^j$.

If j is odd,

$$U_j = \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_z^1\sigma_z^j}$$

will take $\sigma_y^1\sigma_z^2\cdots\sigma_z^{j-1} \to (-1)^{[(j-1)/2]}\sigma_x^1\sigma_z^2\cdots\sigma_z^j$, where $[m/l]$ is the integer part of m/l. The final operator

$$U_{j+1} = \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_y^j}$$

will bring the control operator to the desired one (up to a global phase $(-1)^{[(j-1)/2]}$):

$$\sigma_x^1\sigma_z^2\cdots\sigma_z^{j-1}\sigma_x^j.$$

If j is even, we need an additional unitary operator that flips the first qubit's σ_y^1 into a σ_x^1. This flip is achieved with the operator

$$U_{j+2} = \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\sigma_z^1} = \begin{pmatrix} \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}} & 0 \\ 0 & \mathrm{e}^{\mathrm{i}\frac{\pi}{4}} \end{pmatrix} \otimes \mathbb{1} \otimes \cdots \otimes \mathbb{1}$$

that takes $\sigma_y^1 \to \sigma_x^1$.

Hence, to construct this non-local fermion operator from the standard model requires additional steps that are proportional to j. This number scales polynomially with the complexity so the construction is efficient if the standard model is efficient.

The one and two-body nature of naturally occurring interactions means that a term in a second-quantized representation of a Hamiltonian only has one of two forms: either $a_i^\dagger a_j$ or $a_i^\dagger a_j a_k^\dagger a_l$. We just demonstrated how to handle the first case. The second case merely requires applying that algorithm twice. This squares the complexity.

4.2. State preparation

In this section we discuss the preparation of states of physical relevance. Clearly, the preparation of the initial state is a very important step since the study and efficiency of the given physical process one wants to simulate depends upon it.

Consider a system of N_e fermions and n operators $a_j^\dagger$ (single particle states). A generic N_e-particle state of a Hilbert space $\mathcal{H}_{N_e}$ of antisymmetrized wave functions can always be expanded in terms of the antisymmetric states

$$|\Phi_\alpha\rangle = \prod_{j=1}^{N_e} b_j^\dagger |\mathrm{vac}\rangle,$$

where $b_j^\dagger$ creates a state j and $|\mathrm{vac}\rangle = |0\rangle \otimes |0\rangle \cdots \otimes |0\rangle$ is the vacuum state (i.e. $b_j|0\rangle = 0$, $\forall j$). The operator $b^\dagger$ is in general a linear combination of $a^\dagger$'s, i.e. $b_j^\dagger = \sum_{i=1}^n a_i^\dagger P_{ij}$ where P_{ij} is some matrix and $N_e \leqslant n$.

The states $|\Phi_\alpha\rangle$ $(\alpha = 1, \ldots, \binom{n}{N_e})$ in general form an overcomplete set of non-orthogonal states that span the whole $\mathcal{H}_{N_e}$, i.e. redundantly generate $\mathcal{H}_{N_e}$. They are known as Slater determinants [20]. Typically, $|\Phi_\alpha\rangle$ is the result of a self-consistent mean-field (or generalized Hartree–Fock) calculation. Even a Bardeen–Cooper–Schrieffer superconducting state, which does not preserve the number of particles, can be written in this way after an appropriate canonical transformation which redefines the vacuum state [34].

One can easily prepare the states $|\Phi_\alpha\rangle$ noticing that the quantum gate, represented by the unitary operator

$$U_m = \mathrm{e}^{\mathrm{i}\frac{\pi}{2}(b_m + b_m^\dagger)}$$

when acting on the vacuum state, produces $b_m^\dagger|0\rangle$ up to a phase $\mathrm{e}^{\mathrm{i}\frac{\pi}{2}}$. Therefore, the successive application of similar unitary operators will generate the state $|\Phi_\alpha\rangle$ up to a global phase.

Except for very small systems the total Hilbert space is too large to be fully used (it has an exponential growth with increasing system size). In practice, one works in a subspace of $\mathcal{H}_{N_e}$ that closely represents the physical state one is trying to simulate. Generically, as initial state, we will consider a very general expression of a many-fermion state:

$$\big|\Psi(t=0)\big\rangle = \sum_{\alpha=1}^{N} \mathsf{a}_\alpha |\Phi_\alpha\rangle,$$

where the integer N is a finite and small number. The state can be prepared efficiently (in N) by a number of procedures. We now describe one.

To make the description simple, we will assume $\sum_{\alpha=1}^N |\mathsf{a}_\alpha|^2 = 1$ and $\langle\Phi_\alpha|\Phi_\beta\rangle = \delta_{\alpha\beta}$, which is equivalent to requiring $\{|\Phi_\alpha\rangle\}$ to be an orthonormal set and $|\Psi(t=0)\rangle$ to be normalized to unity. With these assumptions the steps of the state preparation algorithm are:

(1) Adjoin N auxiliary (ancilla) qubits, each in the state $|0\rangle$, to the vacuum of the physical system. The resulting state is

$$\underbrace{|0\rangle \otimes |0\rangle \otimes \cdots |0\rangle}_{N} \otimes |\mathrm{vac}\rangle \equiv |0\rangle_{\mathsf{a}} \otimes |\mathrm{vac}\rangle. \tag{12}$$

(2) From this state generate $\sum_{\alpha=1}^{N} \mathsf{a}_\alpha |\alpha\rangle \otimes |\mathrm{vac}\rangle$ where $|\alpha\rangle$ is an ancilla state with only the α qubit being $|1\rangle$. The procedure for generating this combination of states is described below.

(3) For each $\alpha = 1, \ldots, N$, conditional on the α qubit being $|1\rangle$, apply the state preparation procedure for $|\Phi_\alpha\rangle$. The resulting state is

$$\sum_{\alpha=1}^{N} \mathsf{a}_\alpha |\alpha\rangle \otimes |\Phi_\alpha\rangle. \tag{13}$$

(4) From this state generate

$$\frac{1}{\sqrt{N}} \sum_{\alpha=1}^{N} \mathsf{a}_\alpha |0\rangle_{\mathsf{a}} \otimes |\Phi_\alpha\rangle + \text{terms without } |0\rangle_{\mathsf{a}}. \tag{14}$$

This step will also be described below.

The final step is accepted if a measurement shows all the ancillas being returned to $|0\rangle_{\mathsf{a}}$. The probability of successful preparation is thus $\sum_{\alpha=1}^{N} |\mathsf{a}_\alpha|^2/N = 1/N$. Consequently, on average, N trials will be needed before a successful state preparation.

The procedure to produce step 2 is most easily described by example. We will assume $N = 2$. The problem then is to generate $\mathsf{a}_1|10\rangle \otimes |\mathrm{vac}\rangle + \mathsf{a}_2|01\rangle \otimes |\mathrm{vac}\rangle$ from $|00\rangle \otimes |\mathrm{vac}\rangle$. In what follows all operations will be only on the ancilla part of the initial state so we will not explicitly show the vacuum. We also note that one can always apply a rotation to a given qubit that will take $|0\rangle$ into $x|0\rangle + y|1\rangle$ with $|x|^2 + |y|^2 = 1$. The steps of the procedure are:

(2.1) Adjoin an ancilla qubit $|\mathsf{b}\rangle$ initially being $|0\rangle$. The initial state is now $|0\rangle \otimes |00\rangle$.

(2.2) Conditional on $|\mathsf{b}\rangle = |0\rangle$, rotate the $\alpha = 1$ qubit, and then conditional on the $\alpha = 1$ qubit being $|1\rangle$, flip $|\mathsf{b}\rangle$:

$$x_1|0\rangle \otimes |00\rangle + y_1|1\rangle \otimes |10\rangle. \tag{15}$$

(2.3) Conditional on $|\mathsf{b}\rangle$ being $|0\rangle$, rotate the $\alpha = 2$ qubit, and then conditional on the $\alpha = 2$ qubit being $|1\rangle$, flip $|\mathsf{b}\rangle$:

$$x_1x_2|0\rangle \otimes |00\rangle + x_1y_2|1\rangle \otimes |01\rangle + y_1|1\rangle \otimes |10\rangle. \tag{16}$$

(2.4) Project out the states with $|\mathsf{b}\rangle$ being $|1\rangle$:

$$x_1y_2|01\rangle + y_1|10\rangle. \tag{17}$$

The rotations are chosen so that $\mathsf{a}_1 = y_1$ and $\mathsf{a}_2 = x_1y_2$.

For the explanation of step 4, we will display the physical states. The problem is: Starting with $\mathsf{a}_1|10\rangle \otimes |\Phi_1\rangle + \mathsf{a}_2|01\rangle \otimes |\Phi_2\rangle$, produce (14).

(4.1) Adjoin an ancilla qubit $|\mathsf{b}\rangle$ initially being $|0\rangle$. The initial state is now $\mathsf{a}_1|0\rangle \otimes |10\rangle \otimes |\Phi_1\rangle + \mathsf{a}_2|0\rangle \otimes |01\rangle \otimes |\Phi_2\rangle$.

(4.2) Conditional on $|\mathsf{b}\rangle$ being $|0\rangle$, rotate the $\alpha = 1$ qubit, and then conditional on the $\alpha = 1$ qubit being $|1\rangle$, flip $|\mathsf{b}\rangle$:

$$\mathsf{a}_1\big(x_1|0\rangle \otimes |00\rangle + y_1|1\rangle \otimes |10\rangle\big) \otimes |\Phi_1\rangle + \mathsf{a}_2\big(x_1|0\rangle \otimes |01\rangle + y_1|1\rangle \otimes |11\rangle\big) \otimes |\Phi_2\rangle. \tag{18}$$

(4.3) Conditional on $|\mathsf{b}\rangle$ being $|0\rangle$, rotate the $\alpha = 2$ qubit, and then conditional on the $\alpha = 2$ qubit being $|1\rangle$, flip $|\mathsf{b}\rangle$:

$$\mathsf{a}_1x_1\big(x_2|0\rangle \otimes |00\rangle + y_2|1\rangle \otimes |01\rangle\big) \otimes |\Phi_1\rangle + \mathsf{a}_2x_1\big(x_2|0\rangle \otimes |00\rangle + y_2|1\rangle \otimes |01\rangle\big) \otimes |\Phi_2\rangle. \tag{19}$$

(4.4) Project out the states with $|\mathsf{b}\rangle = |0\rangle$:

$$x_1x_2\big(\mathsf{a}_1|00\rangle \otimes |\Phi_1\rangle + \mathsf{a}_2|00\rangle \otimes |\Phi_2\rangle\big). \tag{20}$$

The rotations are chosen so that x_1x_2 equals $1/\sqrt{N}$ where $N = 2$. Comparing step 2 with step 4, one sees they are structurally identical, differing by the set of amplitudes generated and the complementarity of the subspaces selected for the final result. This latter difference in some sense makes one procedure the inverse of the other. For the case of $N > 2$, one simply replaces steps 2.2 and 2.3 and steps 4.2 and 4.3 by "do loops" over α from 1 to N.

On average, the entire procedure needs N trials before a successful state preparation. (In many cases, the other measurement outcomes can be used also to avoid too many trials.) Construction of the initial

state thus scales as $\mathcal{O}(N^2 n N_e) \leqslant \mathcal{O}(N^2 n^2)$ so unless the number of Slater determinants is exponentially large, general many-fermion states can be initialized efficiently.

4.3. Measurement

While there is a variety of physical observables one measures experimentally and calculates theoretically, at this time it is difficult to demonstrate that they all can be computed efficiently on a quantum computer. Fortunately, we will now argue that one important class of observables, the temporal correlation functions $C_{AB}(t)$, can be computed not only efficiently but also accurately. These functions describe the temporal evolution of some observable $A(t)$ in response to some weak external stimulus that couples to the system's variable $B(0)$. They are at the heart of understanding, for example, the optical properties of materials.

The goal is to determine correlations of the form $C_{AB}(t) = \langle A(t)B(0)\rangle = \langle \mathrm{e}^{\mathrm{i}Ht} A \mathrm{e}^{-\mathrm{i}Ht} B\rangle$ up to a sufficiently small statistical error. Clearly, measuring efficiently C_{AB} is not possible for an arbitrary A and B. One sufficient condition is that A and B are efficiently simulatable Hamiltonians. This observation is based on a method for determining C_{AB} refined by Kitaev [35] and applied to the measurement of correlation functions by Terhal and DiVincenzo [36]. Here we give a different method based on an idea given in [37].

A general principle that can be used to obtain C_{AB} is to decompose the operator whose expectation needs to be determined, i.e. $A(t)B(0)$, into a small sum of operators of a simpler form and measure each summand individually. Our method directly measures expectations of the form $\langle U^\dagger V\rangle$ when algorithms for implementing the unitary operators U and V are available. General correlation functions are then determined by decomposing operators using a unitary operator basis, for example the one consisting of products of Pauli matrices.

The method for measuring $\langle U^\dagger V\rangle$ consists of the following steps:

(1) Adjoin via a direct product an ancilla (i.e. an auxiliary) qubit $\mathbf{a}$ in the state $(|0\rangle + |1\rangle)/\sqrt{2}$ with density matrix $\rho_{\mathbf{a}} = (\mathbb{1} + \sigma_x^{\mathbf{a}})/2$ to the initial state of the system described by the density matrix ρ.

(2) Apply the conditional evolutions $\overline{U}_1 = |0\rangle\langle 0| \otimes U + |1\rangle\langle 1| \otimes \mathbb{1}$ and $\overline{U}_2 = |1\rangle\langle 1| \otimes V + |0\rangle\langle 0| \otimes \mathbb{1}$ ($\overline{U} = \overline{U}_1 \overline{U}_2$). The methods of [24] may be used to implement these evolutions given algorithms for U and V.

(3) Measure $2\sigma_+^{\mathbf{a}} = \sigma_x^{\mathbf{a}} + \mathrm{i}\sigma_y^{\mathbf{a}} = 2|0\rangle\langle 1|$. This may be done by measuring $\sigma_x^{\mathbf{a}}$ and $\sigma_y^{\mathbf{a}}$ in sufficiently many independent trials of these steps.

(4) Given the initial density matrix ρ, the expectation

$$\begin{aligned}\left\langle \sigma_x^{\mathbf{a}} + \mathrm{i}\sigma_y^{\mathbf{a}} \right\rangle_{\rho_{\mathbf{a}}\otimes\rho} &= 2\,\mathrm{Tr}_{n+1}\left[\overline{U}^\dagger |0\rangle\langle 1|\overline{U}\rho_{\mathbf{a}} \otimes \rho\right] \\ &= \mathrm{Tr}_{n+1}\left[|0\rangle\langle 1| \otimes U^\dagger V \rho_{\mathbf{a}} \otimes \rho\right] \\ &= \mathrm{Tr}_n[U^\dagger V \rho] = \langle U^\dagger V\rangle_\rho, \quad (21)\end{aligned}$$

as desired. The statistical noise in the measurement of $\langle U^\dagger V\rangle_\rho$ is determined by that of two binary random variables and therefore depends only on the value of $\mathrm{Tr}_n[U^\dagger V\rho]$, which is inside the unit complex circle. As a result it is a simple matter to determine the number of measurement attempts required to achieve sufficient statistical accuracy.

The procedure for measuring $C_{AB}(t)$ can now be summarized as follows: First express $A = A(0)$ and $B = B(0)$ as a sum of unitary operators $A = \sum_{j=1}^{m_A} A_j$ and $B = \sum_{j'=1}^{m_B} B_{j'}$. A convenient unitary operator basis that works well for the local observables of interest consists of all the products of Pauli operators, as each such product is easily implemented as a quantum algorithm. Then, for each j and j' one uses the just described method with $U = \mathrm{e}^{\mathrm{i}Ht} A_j^\dagger \mathrm{e}^{-\mathrm{i}Ht}$ and $V = B_{j'}$ to obtain $\langle A_j(t)B_{j'}(0)\rangle$. V may be implemented by simulating the evolution under H, applying $B_{j'}$, and then undoing the evolution under H.

An alternative approach to the measurement process is von Neumann's projection method. We sketch it here for completeness and comparison. In this approach we also add an auxiliary (ancilla) degree of freedom to the problem. Suppose that this extra qubit corresponds to an harmonic oscillator degree of freedom $|e\rangle$. Then, we consider the composite state

$$|\Psi\rangle_S \otimes |e\rangle_0,$$

where $|\Psi\rangle_S = \sum_j \lambda_j\ |\phi_j\rangle_S$ is the state of the system we want to probe and $|e\rangle_t$ is the state of the harmonic oscillator in the coordinate (x-) representation. The

corresponding state in the momentum (p-) representation is denoted $|\hat{e}\rangle_t$.

Assume the observable (t-independent Hermitian operator) we want to measure is $\mathcal{A}$. Then, we are interested in determining ${}_S\langle\Psi|\mathcal{A}|\Psi\rangle_S$ in an efficient way. Suppose that we know how to implement the unitary operation $U^S(t) = \mathrm{e}^{-\mathrm{i}\mathcal{A}t}$. Following Kitaev we want to implement the following conditional evolution

$$\mathcal{U} = \sum_t |e\rangle_t\,{}_t\langle e|U^S(t).$$

From the spectral theorem we can write $\mathcal{A} = \sum_j \Lambda_j|\phi_j\rangle_S\,{}_S\langle\phi_j|$. Then,

$$\begin{aligned}\mathcal{U}|\phi_j\rangle_S \otimes |\hat{0}\rangle_0 &= \sum_t \mathcal{U}|\phi_j\rangle_S \otimes |e\rangle_t \\ &= \sum_t \mathrm{e}^{-\mathrm{i}\Lambda_j t}|\phi_j\rangle_S \otimes |e\rangle_t \\ &= |\phi_j\rangle_S \otimes |\hat{\Lambda}_j\rangle_t,\end{aligned}$$

where $|\hat{0}\rangle_0$ is a state with ($p = 0$) zero momentum. Basically, the conditional evolution $\mathcal{U}$ is a momentum translation operator for the harmonic oscillator extra state. Finally,

$$\mathcal{U}|\Psi\rangle_S \otimes |\hat{0}\rangle_0 = \sum_j \lambda_j|\phi_j\rangle_S \otimes |\hat{\Lambda}_j\rangle_t.$$

Although the second measurement method is conceptually simpler, it requires approximately implementing the ancillary harmonic oscillator, the conditional evolutions for many different choices of t, and a more complex analysis of the measurement statistics. The conditional evolutions can be simplified somewhat, and in special cases (such as as a subroutine of factoring) become very efficient—see [35].

4.4. Measurement noise control

The quantum physics simulation algorithm described above is approximate and the output is noisy. In order to properly use it, we need to have explicit estimates of the error ϵ in the inferred expectations given the noise in the implementation. Furthermore, the effort required to make ϵ small must scale polynomially with $1/\epsilon$. There are three sources of error that need to be considered. The first is associated with intrinsic noise in the implementation of the gates due to imperfections and unwanted interactions. The second comes from the discretization of the evolution operator and the use of the Trotter decomposition. The third is due to the statistics in measuring the desired correlation function using the technique given above.

4.4.1. Gate imperfections

The problem of gate imperfections can be dealt with by using quantum error correction [38,39] and fault tolerant quantum computation [40–44]. According to the accuracy threshold theorem, provided the physical gates have sufficiently low error, it is possible to quantum compute arbitrarily accurately. The fault tolerant computation implements unitary operations and measurements on encoded qubits with overheads bounded by $\mathcal{O}(\log^k(1/\epsilon))$ for some k. This exponentially efficient convergence implies that the effects of physical noise can in principle be ignored.

4.4.2. Discretization error

A second type of error is the one introduced by the discretization of the evolution operator. This discretization is very similar to the one used in classical simulation of dynamical quantum systems. It is possible to estimate the size of this error by a detailed analysis of the discretization. For example, using the Trotter approximation

$$\begin{aligned}\mathrm{e}^{-\mathrm{i}(H_1+H_2)\Delta t} &= \mathrm{e}^{-\mathrm{i}H_1\Delta t/2}\mathrm{e}^{-\mathrm{i}H_2\Delta t}\mathrm{e}^{-\mathrm{i}H_1\Delta t/2} \\ &\quad + \mathcal{O}\big((\Delta t)^3\big).\end{aligned}$$

The coefficient of $(\Delta t)^3 \sim -\mathrm{i}(H_1 + H_2)^3/6$ can be bounded by estimating the largest eigenvalue of H_1 and H_2.

4.4.3. Measurement statistics

Our technique for measuring the correlation function $\langle A(t)B(0)\rangle$ requires measuring the expectations of unitary operators $U_j^\dagger V_{j'}$ associated with the implemented evolution. In most cases, the operators A and B are a sum of $\mathcal{O}(m_{A,B})$ products of Pauli matrices, so that $\mathcal{O}(m_A)$ $U_j^\dagger$'s and $\mathcal{O}(m_B)$ $V_{j'}$'s are needed. This means that the expectation is a sum of $\mathcal{O}(m_A m_B)$ random variables $r_{jj'}$, where $|r_{jj'}| \leqslant 1$. To assure that the statistical noise (given by the standard deviation) is less than ϵ it suffices to measure each $r_{jj'}$ $\mathcal{O}(m_A m_B/\epsilon^2)$ times.

5. Resonant impurity scattering

5.1. Formulation of the physical problem

Our toy problem is a ring of n equally-spaced lattice sites on which spinless fermions hop to nearest neighbor sites or hop onto or from an "impurity" state. The length of the ring is $L = na$, where a is the distance between sites. The system is described by the Hamiltonian (in second quantized form)

$$H = -\mathcal{T}\sum_{i=1}^{n}\left(c_i^\dagger c_{i+1} + c_{i+1}^\dagger c_i\right) + \epsilon b^\dagger b + \frac{V}{\sqrt{n}}\sum_{i=1}^{n}\left(c_i^\dagger b + b^\dagger c_i\right), \tag{22}$$

where $\mathcal{T}$ is the hopping matrix element, ϵ is the energy of the localized (impurity) state, and V is a hybridization energy. As usual, b's and c's are fermion (anticommuting) operators. The index i labels the lattice sites ($R_i = ia$ is the lattice site position) and strict periodic boundary conditions are assumed, i.e.

$$c_{i+n}^\dagger = c_i^\dagger. \tag{23}$$

We now imagine that the system is initially prepared in the zero temperature ground state of the ring in the absence of the impurity. Then, at time $t = 0$, a fermion is injected into the impurity state. After the system has evolved for some time t, we want to compute the probability amplitude that the evolved state is still in the initial state. The relevant quantity to compute is ($\hbar = 1$ and $t \geqslant 0$)

$$G(t) = \left\langle \Psi(0)\middle| b(t)b^\dagger(0)\middle|\Psi(0)\right\rangle, \tag{24}$$

$$b(t) = \mathrm{e}^{\mathrm{i}Ht} b(0) \mathrm{e}^{-\mathrm{i}Ht}, \tag{25}$$

where the initial state is the Fermi sea of $N_e \leqslant n$ fermions

$$\left|\Psi(0)\right\rangle = |FS\rangle = \prod_{i=0}^{N_e-1} c_{k_i}^\dagger |0\rangle. \tag{26}$$

$|0\rangle$ is the vacuum of fermions and

$$c_{k_i}^\dagger = \frac{1}{\sqrt{n}}\sum_{j=1}^{n} \mathrm{e}^{\mathrm{i}k_i R_j} c_j^\dagger. \tag{27}$$

The wave number k_j is determined from the periodic boundary conditions, $c_{i+n}^\dagger = c_i^\dagger$, which implies

$$k_j = \frac{2\pi n_j}{L}, \quad \text{with } n_j \text{ an integer.} \tag{28}$$

There is no unique way to choose the set of n_j's. The common convention is to define the first Brillouin zone as

$$-\frac{\pi}{a} < k \leqslant \frac{\pi}{a}, \tag{29}$$

with k values uniformly distributed in this interval with spacing $\Delta k = 2\pi/L$.

5.2. Quantum algorithm

We want to write a quantum algorithm that allows one to compute $G(t)$. To this end, we start by representing fermion operators in terms of Pauli matrices. Because of the form of the hybridization term a most convenient representation is the following

$$\begin{aligned} b^\dagger &= \sigma_+^1, \\ c_{k_0}^\dagger &= -\sigma_z^1\sigma_+^2, \\ &\vdots \\ c_{k_{n-1}}^\dagger &= (-1)^n \sigma_z^1\sigma_z^2\cdots\sigma_z^n\sigma_+^{n+1}, \end{aligned}$$

from which the following mapping results

$$\begin{aligned} b^\dagger b &= \tfrac{1}{2}\left(\mathbb{1} + \sigma_z^1\right), \\ c_{k_i}^\dagger c_{k_i} &= \tfrac{1}{2}\left(\mathbb{1} + \sigma_z^{i+2}\right), \\ c_{k_0}^\dagger b + b^\dagger c_{k_0} &= \tfrac{1}{2}\left(\sigma_x^1\sigma_x^2 + \sigma_y^1\sigma_y^2\right). \end{aligned}$$

Therefore, the Hamiltonian operator reads

$$2H = \left[\epsilon + \sum_{i=0}^{n-1}\mathcal{E}_{k_i}\right]\mathbb{1} + \epsilon\sigma_z^1 + \sum_{i=0}^{n-1}\mathcal{E}_{k_i}\sigma_z^{i+2} + V\left(\sigma_x^1\sigma_x^2 + \sigma_y^1\sigma_y^2\right), \tag{30}$$

where $\mathcal{E}_k = -2\mathcal{T}\cos ka$. An additional simplification can be introduced when one realizes that the structure of the observable to be measured is such that

$$b(t) = \mathrm{e}^{\mathrm{i}Ht} b(0) \mathrm{e}^{-\mathrm{i}Ht} = \mathrm{e}^{\mathrm{i}\overline{H}t}\sigma_-^1\mathrm{e}^{-\mathrm{i}\overline{H}t}, \tag{31}$$

where $\overline{H}$ is given by

$$\overline{H} = \frac{\epsilon}{2}\sigma_z^1 + \frac{\mathcal{E}_{k_0}}{2}\sigma_z^2 + \frac{V}{2}\left(\sigma_x^1\sigma_x^2 + \sigma_y^1\sigma_y^2\right), \tag{32}$$

and, therefore, the "string" one has to simulate has length equal to two (it involves only qubits 1 and 2)

$$\mathcal{A}(t) = b(t)b^\dagger(0) = \mathrm{e}^{\mathrm{i}\overline{H}t}\sigma_-^1\mathrm{e}^{-\mathrm{i}\overline{H}t}\sigma_+^1. \tag{33}$$

If we were to transform $\overline{H} = U H_{P1} U^\dagger$ unitarily with $U = \prod_{j=1}^{n} \mathrm{e}^{\mathrm{i} H_{P2}^{j} t_j}$ and n a finite integer ($U U^\dagger = \mathbb{1}$) in such a way that both H_{P1} and H_{P2} are physical Hamiltonians, then the simulation would be straightforward. (We call this type of mapping a physical unitary mapping.) For our two qubit case, one can always perform a physical unitary mapping with

$$U = \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_x^2} \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\sigma_y^1} \mathrm{e}^{-\mathrm{i}\frac{\theta}{2}\sigma_z^1\sigma_z^2} \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_y^1} \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_x^1} \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\sigma_x^2} \times \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\sigma_y^2} \mathrm{e}^{\mathrm{i}\frac{\theta}{2}\sigma_z^1\sigma_z^2} \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\sigma_x^1} \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_y^2}, \tag{34}$$

$$H_{P1} = \tfrac{1}{2}\big(E - \sqrt{\Delta^2 + V^2}\big)\sigma_z^1 + \tfrac{1}{2}\big(E + \sqrt{\Delta^2 + V^2}\big)\sigma_z^2, \tag{35}$$

with $E = (\epsilon + \mathcal{E}_{k_0})/2$, $\Delta = (\epsilon - \mathcal{E}_{k_0})/2$, and $\cos\theta = 1/\sqrt{1+\delta^2}$ with $\delta = (\Delta + \sqrt{\Delta^2 + V^2})/V$.

In general, such a constrained transformation is not easily realized and one performs a Trotter decomposition

$$\mathrm{e}^{\mathrm{i}\overline{H}t} = \big[\mathrm{e}^{\mathrm{i}\overline{H}s}\big]^M = \big[\mathrm{e}^{\mathrm{i}\overline{H}_z s} \mathrm{e}^{\mathrm{i}\overline{H}_{xy} s} + \mathcal{O}(s^2)\big]^M \tag{36}$$

where $\overline{H} = \overline{H}_z + \overline{H}_{xy}$ with $\overline{H}_{xy} = \frac{V}{2}(\sigma_x^1\sigma_x^2 + \sigma_y^1\sigma_y^2)$ and time slice $s = t/M$. On the other hand, one can easily perform a physical unitary mapping for $\mathrm{e}^{\mathrm{i}\overline{H}_{xy}s}$

$$\mathrm{e}^{\mathrm{i}\overline{H}_{xy}s} = \overline{U} \mathrm{e}^{\mathrm{i}H_{P1}s} \overline{U}^\dagger, \tag{37}$$

where $H_{P1} = \frac{V}{2}(\sigma_x^1 - \sigma_y^2)$ and

$$\overline{U} = \mathrm{e}^{\mathrm{i}\frac{\pi}{4}\sigma_x^2} \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\sigma_y^1} \mathrm{e}^{-\mathrm{i}\frac{\pi}{4}\sigma_z^1\sigma_z^2}. \tag{38}$$

Finally, the "string" one has to simulate with the quantum computer is

$$\mathcal{A}(t) \simeq \big[S(s)\big]^M \sigma_-^1 \big[S^\dagger(s)\big]^M \sigma_+^1, \quad S(s) = \mathrm{e}^{\mathrm{i}\overline{H}_z s} \overline{U} \mathrm{e}^{\mathrm{i}H_{P1}s} \overline{U}^\dagger, \quad \text{and } G(t) = \langle \mathcal{A}(t) \rangle. \tag{39}$$

6. Concluding remarks

We investigated the implementation of algorithms for the simulation of fermionic quantum systems, and gave an explicit mapping that relates the usual qubit of a quantum computer to the fermionic modes that we want to simulate. Our attention focused on the so-called sign problem. It is a problem appearing in attempts to simulate classically the dynamics of quantum systems. We reviewed the origin of this problem and showed how this problem is avoided in quantum computing simulation. The evolution of quantum computers are intrinsically quantum mechanical and this is the main difference with a classical computer that allows one to solve the sign problem. We studied sources of errors in a quantum computer, such as gate imperfections and the expansion of the evolution operator, and argued that they would not open a back door to a problem similar to the sign problem.

We gave a very general definition of what a model of quantum computation is. In particular and because of our particular interest, i.e. the simulation of fermion systems, we described the standard and the fermionic models ("Grassmann Chip"). These are, of course, not the only ones. Isomorphisms of $*$-algebras allow one to introduce more "esoteric" models [9]. Indeed, there is nothing special about the spinless fermionic model of quantum computation. One could have used a "hard-core boson" model which admits, in principle, a realization in terms of He^4 atoms. The key point is the implementation of the physical gates.

Our effort focused on the simulation of the dynamics of fermionic quantum systems. However other problems can be of interest: the thermodynamic or ground state properties of a Hamiltonian. Even if one had a quantum computer, it is not clear how to use it to efficiently compute these quantities. On the other hand, at present, no proof exists showing that this is not possible.

An approach that in principle could be used to compute the spectrum of a Hamiltonian H (e.g., the ground state) or expectation values of arbitrary observables is the adiabatic "switching on" in conjunction with the Gell–Mann–Low theorem [45] of quantum field theory. The idea simply consists of introducing a fictitious Hamiltonian

$$H_\epsilon(t) = H_0 + f_\epsilon(t) H_1, \tag{40}$$

where both H_0 and H_1 are time independent operators ($H = H_0 + H_1$) and the scalar function $f_\epsilon(t)$ is such that $\lim_{t\to\pm\infty} f_\epsilon(t) = 0$ and $\lim_{t\to 0} f_\epsilon(t) = 1$, for an arbitrary adiabatic parameter ϵ. In other words, $H_\epsilon(t=0) = H$ and $H_\epsilon(t = \pm\infty) = H_0$. H_0 is typically an operator whose spectrum is known, e.g., an arbitrary bilinear operator representing a mean-field solution of H and whose eigenstates can be

straightforwardly prepared (let's call it $|\Phi_0\rangle$). The Gell–Mann–Low theorem asserts that

$$\lim_{\epsilon\to 0}\frac{U_\epsilon(0,-\infty)|\Phi_0\rangle}{\langle\Phi_0|U_\epsilon(0,-\infty)|\Phi_0\rangle}=\frac{|\Psi_0\rangle}{\langle\Phi_0|\Psi_0\rangle} \tag{41}$$

if the state whose limit one is performing admits a series expansion in a coupling parameter characterizing the strength of H_1. This formal device generates the eigenstate adiabatically connected to $|\Phi_0\rangle$. The theorem does not guarantee that if $|\Phi_0\rangle$ is the ground state of H_0 then $|\Psi_0\rangle$ is the ground state of H. If the conditions of the theorem are satisfied then computation of the spectrum of H is straightforward. To our knowledge this approach has never been implemented in practice.

The work presented here is only a first step in a program investigating the simulation of quantum systems using quantum computers. We have given a rather explicit algorithm for a simple problem and we will increase the complexity of the problems in the work to come. An interesting problem would be to provide algorithms to test for superconductivity in systems such as the Hubbard model. Such simulations using classical computers cannot unequivocally answer this important question because of the sign problem, but a quantum computer could.

References

[1] P.W. Shor, SIAM J. Comput. 26 (1997) 1484.
[2] L.K. Grover, Phys. Rev. Lett. 79 (1997) 325.
[3] R.P. Feynman, Int. J. Theor. Phys. 21 (1982) 467.
[4] D.S. Abrams, S. Lloyd, Phys. Rev. Lett. 79 (1997) 2586.
[5] M.H. Kalos, D. Levesque, J. Verlet, Phys. Rev. A 9 (1974) 2178.
[6] D.M. Ceperley, Rev. Mod. Phys. 67 (1995) 279.
[7] D.M. Ceperley, in: E. Schachinger, H. Mitter, M. Sormann (Eds.), Recent Progress in Many-Body Theories, Vol. 4, Plenum Press, New York, 1995, pp. 455–470.
[8] In two spatial dimensions, other possibilities for quantum statistics emerge. For example, fractional statistics for particles called anyons interpolates continuously between bosons and fermions.
[9] C.D. Batista, G. Ortiz, Phys. Rev. Lett. 86 (2001) 1082.
[10] From the point of view of complexity theory, an algorithm is efficient if it scales polynomially in time.
[11] G. Ortiz, D.M. Ceperley, R.M. Martin, Phys. Rev. Lett. 71 (1993) 2777.
[12] J. Bonča, J.E. Gubernatis, Phys. Rev. E 53 (1996) 6504.
[13] M. Jarrell, J.E. Gubernatis, Phys. Rep. 269 (1996) 133.
[14] D. Deutsch, Proc. Roy. Soc. London A 400 (1985) 97.
[15] We thank E. Fradkin for providing us with a preprint that circulated on April's fool day in 1983, where the term Grassmann chip is used as a joke. The manuscript is "Monte Carlo Simulation of a Realistic Unified Gauge Theory", by A. Chodos and J. Rabin (unpublished).
[16] L. Onsager, Phys. Rev. 65 (1944) 117.
[17] S. Bravyi, A. Kitaev, quant-ph/0003137, unpublished.
[18] R.P. Feynman, A.R. Hibbs, Quantum Mechanics and Path Integrals, McGraw-Hill, New York, 1965.
[19] The system is composed of N_e particles moving in d spatial dimensions ($\hbar = m = e = 1$), and a generic point in a flat Cartesian manifold of dimension $D = dN_e$ is represented by $\mathcal{R} = (\mathbf{r}_1, \ldots, \mathbf{r}_{N_e})$. $V(\mathcal{R})$ is the potential energy operator and p_i is particle's i canonical momentum.
[20] J.W. Negele, H. Orland, Quantum Many-Particle Systems, Addison-Wesley, Redwood City, 1988.
[21] W. von der Linden, Phys. Rep. 220 (1992) 53.
[22] S. Lloyd, Science 273 (1996) 1073;
D.A. Meyer, J. Stat. Phys. 85 (1996) 551;
S. Wiesner, quant-ph/9603028;
B.M. Boghosian, W. Taylor IV, Physica D 120 (1998) 30;
C. Zalka, Proc. Roy. Soc. London A 454 (1998) 313.
[23] E. Knill, R. Laflamme, quant-ph/9909094, unpublished.
[24] A. Barenco et al., Phys. Rev. A 52 (1995) 3457.
[25] D. DiVincenzo, Phys. Rev. A 51 (1995) 1015.
[26] From the theory of computation point of view it is necessary to make additional assumptions on how the functions may be prescribed. In particular the functions themselves must be classically computable in a suitable sense. This problem is avoided by permitting only a finite set of quantum gates instead of continuously controllable Hamiltonians.
[27] A. Peres, Quantum Theory: Concepts and Methods, Kluwer Academic Publishers, Dordrecht, 1998.
[28] E. Bernstein, U. Vazirani, SIAM J. Comput. 26 (1997) 1411.
[29] R. Cleve, quant-ph/9906111, unpublished.
[30] D. Aharonov, quant-ph/9812037, unpublished.
[31] P. Jordan, E. Wigner, Z. Phys. 47 (1928) 631.
[32] E. Fradkin, Phys. Rev. Lett. 63 (1989) 322.
[33] L. Huerta, J. Zanelli, Phys. Rev. Lett. 71 (1993) 3622.
[34] M. Guerrero, G. Ortiz, J.E. Gubernatis, Phys. Rev. B 59 (1999) 1706.
[35] A.Y. Kitaev, quant-ph/9511026.
[36] B.M. Terhal, D.P. DiVincenzo, Phys. Rev. A 61 (2000) 2301.
[37] E. Knill, R. Laflamme, Phys. Rev. Lett. 81 (1998) 5672.
[38] P.W. Shor, Phys. Rev. A 52 (1995) 2493.
[39] A. Steane, Proc. Roy. Soc. London A 452 (1996) 2551.
[40] P.W. Shor, in: Proceedings of the Symposium on the Foundations of Computer Science, IEEE Press, Los Alamitos, California, 1996, pp. 56–65.
[41] D. Aharonov, M. Ben-Or, in: Proceedings of the 29th Annual ACM Symposium on the Theory of Computation (STOC), ACM Press, New York, 1996, pp. 176–188.
[42] A.Y. Kitaev, in: O. Hirota, A.S. Holevo, C.M. Caves (Eds.), Quantum Communication, Computing and Measurement, Plenum Press, New York, 1997.
[43] E. Knill, R. Laflamme, W.H. Zurek, Science 279 (1998) 342.
[44] J. Preskill, Proc. Roy. Soc. London A 454 (1998) 385.
[45] M. Gell-Mann, F. Low, Phys. Rev. 84 (1951) 350.

ELSEVIER

Computer Physics Communications 146 (2002) 317–323

Computer Physics Communications

www.elsevier.com/locate/cpc

Lattice Boltzmann schemes for quantum applications

Sauro Succi

Istituto Applicazioni Calcolo, IAC-CNR, Viale Policlinico 137, 00161, Roma, Italy

Received 29 January 2001

Abstract

We review the basic ideas behind the quantum lattice Boltzmann equation (LBE), and present a few thoughts on the possible use of such an equation for simulating quantum many-body problems on both (parallel) electronic and quantum computers.

PACS: 02.70.-c; 03.65-w; 03.75.-b

1. Quantum mechanics and fluids

Intriguing analogies between quantum mechanics and fluid mechanics have been pointed out since the earliest days of quantum theory [1]. The orthodox tenet is that these analogies are purely formal in character and do not bear upon the basic physics of quantum phenomena. A less-orthodox, albeit not minor, stream of thought insists instead that quantum mechanics, and notably Heisenberg's uncertainty principle, are nothing but a mirror of our ignorance of the underlying (hidden) microscopic physical level. This leads to the puzzling theory of 'hidden variables' which traces back to Einstein and subsequently to Bohm and others [2]. It is not our intent here to enter this fascinating and still open subject [3]. We turn to a practical question instead: what can the analogy do for us in terms of numerical modeling of evolutionary quantum mechanical phenomena? The question is legitimate because, regardless of its philosophical implications, the fluid analogy certainly provides an intuitive and physical sound basis to develop numerical methods for time-dependent quantum mechanics. In particular, it is reasonable to ask whether the advantages brought about by lattice kinetic methods in fluid dynamics can—by means of the fluid analogy—be exported to the context of quantum mechanics. Before we put forward our discrete kinetic theory version of the analogy, it is useful to provide a cursory survey of the main ideas behind the analogy itself. To this end, a short recap of basic notions of quantum mechanics is in order.

2. The fluid formulation of the Schrödinger equation

Let us begin with the Schrödinger equation for a non-relativistic quantum particle of mass m in an external potential $V(\vec{x})$:

$$i\hbar\partial_t\Psi = \left[-\frac{\hbar^2}{2m}\Delta + V(\vec{x})\right]\Psi, \tag{1}$$

where $\Psi(\vec{x},t)$ is the wavefunction of the material particle. Upon multiplying (1) by the complex conjugate

E-mail address: succi@iac.rm.cnr.it (S. Succi).

0010-4655/02/$ – see front matter
PII: S0010-4655(02)00422-8

Ψ^*, and the complex conjugate of (1) by Ψ and then subtracting, we obtain the following set of fluid equations:

$$\partial_t \rho + \partial_a J_a = 0, \tag{2}$$

$$\partial_t J_a + \partial_a P = 0, \tag{3}$$

where, by using the eikonal representation $\Psi = \rho^{1/2} e^{i\theta}$:

$$\rho = |\Psi|^2, \tag{4}$$

$$J_a = \frac{\hbar}{m}\partial_a \theta \equiv \rho u_a, \tag{5}$$

$$P = \rho\left(\frac{u^2}{2} + \frac{V}{m} + \frac{V^Q}{m}\right), \tag{6}$$

and

$$V^Q \equiv -\frac{\hbar^2}{2m}\frac{\Delta\rho^{1/2}}{\rho^{1/2}} \tag{7}$$

is the famous quantum potential advocated by Bohm and coworkers to support the picture of quantum mechanics as an intrinsically non-local description of the microscopic world [4]. This configures quantum matter as an *ideal (inviscid, dissipationless), irrotational compressible* fluid. The inviscid character of the quantum fluid stems from the reversible nature of the Schrödinger equation, a diffusion equation in *imaginary* time. So much for the analogy in the continuum. What about the discrete lattice world? Interestingly enough, this analogy becomes even *more* compelling once transposed into the language of the lattice world. In fact, the lattice formulation naturally calls for an an "upgrade" from the non-relativistic Schrödinger equation to its relativistic associate, the Dirac equation. Symbolically, the analogy goes as follows (*DE*: Dirac equation, *SE*: Schrödinger equation, *LBE*: Lattice Boltzmann equation, *NSE*: Navier–Stokes equation):

$$DE \to SE, \tag{8}$$

$$LBE \to NSE. \tag{9}$$

2.1. Fluid formulation of relativistic quantum mechanics

To unfold this analogy, it proves expedient to cast the Dirac equation into a form where all streaming matrices, known as Weil matrices, become real. This is the so-called *Maiorana form*: In a compact four-dimensional notation, this reads

$$\left[W^\mu_{jk}\partial_\mu\right]\psi_k = iM_{jk}\psi_k, \quad \mu = 0,3 \tag{10}$$

with

$$W^0_{jk} = \delta_{jk},$$

$$W^1_{jk} = \alpha^x_{jk}, \quad W^2_{jk} = \beta_{jk}, \quad W^3_{jk} = -\alpha^z_{jk},$$

$$M_{jk} = -im\alpha^y_{jk} + qV\delta_{jk} + A^a_{jk}J_a,$$

where all matrices have the standard meaning. Here $J^\mu A_\mu \equiv qV + J_a A^a$ is the interaction of the elementary charge q with an external electromagnetic field described by the 4-vector potential (V, A^a).

A scalar product of Eq. (10) with ψ^*_j yields the desired set of continuity equations:

$$\partial_t \rho_j + \partial_a J^a_j = S_j, \quad j = 1,4 \tag{11}$$

where $\rho_j = \psi^*_j\psi_j$ is the partial density of the jth fluid, $J^a_j = \psi^*_j\alpha^a_{jk}\psi_k$ the corresponding current density, and $S_j = i\psi^*_j M_{jk}\psi_k$ is a "chemical" source term transferring mass across the different components of the relativistic mixture. Note that in the above expressions only the index k is summed upon. Unitarity, read norm conservation, implies $\sum_j S_j = \sum_{jk}\psi^*_j M_{jk}\psi_k = 0$. This is automatically secured by the antihermitian character of the mass matrix: $M_{kj} + M^*_{jk} = 0$. As promised, the fluid analogy comes by *more* naturally than in the non-relativistic case, because the Dirac equation only involves first order derivatives. Another pleasing feature is that the external interaction is easily accommodated into a formal redefinition of the mass matrix, without compromising the local nature of the theory. The fluid interpretation of the Dirac equation is equally transparent: four types of *spinning particles* stream in space and, once on the same space-time location, they interact via the "scattering matrix" M_{jk}. A qualitative difference with classical particle motion is apparent, though. In a classical fluid, particles do not "mix" during the streaming phase. A type-1 particle at location x at time t with speed v propagates to $x + v\,dt$ at time $t + dt$ and it is still entirely of type 1.

A relativistic particle however undergoes mixing during free propagation, because its spinning motion implies a rotation around the direction of motion which mixes up the four spinorial components. This is why the streaming matrix is generally non-diagonal, echoing the fact that spin is not an ordinary vector. This suggests that the discrete space-time of a relativistic particle should be represented by a 'hypernet-

ted lattice' in which each link is made up of four distinct but communicating channels, one per spinorial state. This "Hypernetted Lattice Theory" is less of a joke than it seems. It has been recently realized that lattice formulations of field theory based upon spinning particle motion may offer potential advantages over more popular techniques such as path-integration [5]. This is because in quantum lattice models "*instead of seeking discretized versions of the Hamiltonian or the Lagrangian, a discretized version of the evolution operator is introduced*" [5]. In fact, what this author finds is that "*the rotation group, the Lorentz group and spin emerge automatically in the continuum limit from unitary dynamics on a cubic lattice*". The reader fond of more details is directed to the original reference.

2.2. Dirac to Schrödinger: the adiabatic approximation

As noted in [6], the way the Schrödinger equation is obtained as a long wavelength (low energy) limit of the Dirac equation involves a sort of adiabatic approximation which is formally very similar to the low-Knudsen adiabatic expansion taking the (lattice) Boltzmann equation into the Navier–Stokes equations. The formal parallel emerging from this analogy is

$$Kn = l_\mu / l_M \sim \beta = v/c, \tag{12}$$

where l_μ is the particle mean free path, l_M a typical coherence length of the macroscopic fluid and β is the relativistic particle to light speed ratio. Detailed calculations can be found in the original reference [6] and need not be repeated here.

The relativistic motion implies that any particle of momentum p_a is invariably associated with an antiparticle with opposed momentum $-p_a$. The symmetric combination of these two gives rise to a smooth, emergent field, ϕ^+, whereas the antisymmetric combination defines a low amplitude, high-frequency mode which decouples from the system dynamics in the limit $\beta \to 0$. The scenario is exactly the same as the adiabatic approximation in kinetic theory, with a key difference. Kinetic theory describes dissipative phenomena in which adiabatic elimination wipes out the initial conditions, the transient modes die out, never to return. Quantum mechanics is reversible, and fast modes never die out: they just oscillate so fast that any observation on timescales longer than their period of oscillation simply overlooks them. But they are still there and more resolved (higher energy) measurements could always bring them back again. Note that it is the fast mode, not the antiparticle mode that fades away; the particle–antiparticle twin-link does not dissolve even in the low energy limit.

Another interesting remark concerns the symmetry breaking induced by a non-zero mass m. If m is made zero the up and down walkers do not see each other and go across with no interaction, the result being the wave equation for photons. Manifestly this is a singular limit which cannot be described by the Schrödinger equation (diffusion coefficient goes to infinity). Any non zero mass causes "collisions" which slow down the wavepackets and confer them a subluminal speed $v < c$ as it befits material particles.

2.3. The interacting case

Interactions with an external or self-consistent fields are readily included by a minor extension of the "collision operator". They read as follows:

$$\partial_t u_{1,2} - \partial_z u_{1,2} = m d_{2,1} + \mathrm{i} g d_{2,1}, \tag{13}$$

$$\partial_t d_{1,2} + \partial_z d_{1,2} = -m u_{2,1} + \mathrm{i} g u_{1,2}, \tag{14}$$

where $g = eV/\hbar$ is the coupling frequency of the potential. Self-consistent potentials, such as those arising in connection with the non-linear Schrödinger equation, are easily accommodated by making g a function of the local density $u^2 + d^2$.

3. The quantum lattice Boltzmann equation

We are finally in the position to reformulate the basic analogy in quantitative terms. This is based on the following position: *The 4-spinor* $\psi_j(\vec{x}, t) \equiv \psi(\vec{x}, \vec{s}_j, t)$ *is identified with a complex discrete particle distribution* $f_i(\vec{x}, t) \equiv f(\vec{x}, \vec{v}_i, t)$. The analogy is tantalizing, but a minute's thought reveals two severe flaws:

(1) While the 4-spinor ψ_j (we consider spin 1/2 throughout) has always four components in any dimensions, the discrete population f_i is a set of b real functions with b a sensitive function of space dimensionality.

(2) While LBE streaming is always diagonal in momentum space, the three Weil matrices can*not* be simultaneously diagonalized.

Both problems are intimately related to the quantum nature of the spin variable. Fortunately, there is a way out. As observed in [6] in the limit of 'small' timesteps, actually much shorter than the inverse Compton frequency ω_c^{-1}, *both flaws can be circumvented by decomposing the three-dimensional particle motion into a sequence of three one-dimensional motions along the coordinate directions* x, y, z. The technical key to achieve this task is a well known tool-of-the-trade in computational fluid dynamics: "Operator Splitting". The main use of operator splitting in Computational Fluid Dynamics is to handle 3D problems as a sequence of lower dimensional ones. In quantum field theory, a very similar technique goes under a different name: "Trotter formula": $e^A = (e^{A/n})^n$ with n integer and A any 'reasonable' operator. Consider the formal solution to the Dirac equation for a massless particle (the collisional operator plays no role at this stage):

$$\Psi_j\left(x^\mu + \mathrm{d}x_\mu\right) = \left[e^{\mathrm{d}t\sum_{\mu=0}^{3} W_{jk}^\mu \partial_\mu}\right]\Psi_k(x^\mu). \tag{15}$$

Manifestly, the propagator taking the wavefunction from $x^\mu \equiv (x, y, z, t)$ to $x^\mu + \mathrm{d}x^\mu \equiv (x + \mathrm{d}x, y + \mathrm{d}y, z + \mathrm{d}z, t + \mathrm{d}t)$ is the direct product of three one-dimensional partial propagators $P^a \equiv e^{\mathrm{d}t[\partial_t + W_{jk}^a \partial_a]}$, $a = x, y, z$ (no summation upon a implied). This is the natural consequence of the additivity of the streaming operator. This expression is a good starting point for "conventional" numerical treatment of the Dirac equation [7], but is definitely unsuitable to a quantum LBE formulation because spinorial states get mixed during the propagation step, something that would not occur to a classical particle.

Therefore, a naive application of operator splitting is not viable.

However, we can argue that we *do not* need to work with the same representation of the Dirac equation during the three separate streaming steps. As long as we are able to develop a recipe securing uniqueness of the representation in x_μ and $x_\mu + \mathrm{d}x_\mu$, we are free of choosing the representation that better fits our needs. The idea is to perform each 1D partial streaming in the representation where the corresponding Weil matrix is diagonal. In practice, one propagates along one direction, say x, then 'rotates' the system so as to diagonalize the Weil matrix along, say, y, so that propagation along y can be performed like for a classical particle, and finally 'rotates back' the propagated solution at $(x + \mathrm{d}x, y + \mathrm{d}y, t + \mathrm{d}t)$. New errors are introduced in the numerical treatment, but we shall argue that they are $O(\mathrm{d}t^2)$, namely within the general accuracy of the LBE method.

The quantum LBE bears many similarities with other quantum lattice schemes discussed in the recent [5,8,9] and not so recent [10] literature. What sets it apart from all these schemes is the fact of insisting on a diagonal representation of the Weil matrices, so as *to retain the notion of classical trajectories as much as we can*. In fact, the "turn" operator R can formally be interpreted as an "internal scattering" between particle–antiparticle states [9], thus leaving the concept of quantum trajectory still well defined, although in a generalized sense. In a pictorial sense [10], we might say that while classical particles just "Stream and Collide", quantum particles, like swimmers, need a somersault before they can turn in space: they "Stream, Turn and Collide"! The 'Turn' step is a necessity induced by the internal structure of the relativistic particle.

Leaving the details to the original work, here we simply report the final result for a pair of 'up' and 'down' walkers in one-dimension. Upon using a Cranck–Nicholson time-marching procedure (securing unitarity of the numerical scheme), the quantum LBE takes the following form:

$$u(z + \mathrm{d}z, t + \mathrm{d}t) = Au(z, t) + Bd(z, t), \tag{16}$$

$$d(z - \mathrm{d}z, t + \mathrm{d}t) = Ad(z, t) - Bu(z, t), \tag{17}$$

where

$$A = \frac{1 - \Omega/4}{1 - \Omega/4 - \mathrm{i}g}, \tag{18}$$

$$B = \frac{m}{1 - \Omega/4 - \mathrm{i}g}, \tag{19}$$

$$\Omega = m^2 - g^2. \tag{20}$$

A few comments are in order.

First, with $g = 0$ (no-interaction), implicit time marching translates into a mere redefinition of the particle mass $m \to m' = m/(1 - m^2/4)$. By reinstating the time-step Δt, it is easily recognized that

$m' \to m$ in the limit $\Delta t \to 0$, which means that quantum LBE fulfills the requirement of numerical *consistency*. Large timesteps $m\Delta t > 1$ lead to unphysical results, as it is to be expected since the natural Compton frequency m (in atomic lattice units $\hbar = c = \Delta t = \Delta x = 1$), is no longer resolved. Simple algebra also shows that quantum LBE is *unconditionally stable and* norm-preserving (the all-important unitarity condition). This is fairly remarkable for an explicit numerical scheme [11], and ultimately traces back to the (implicit) lightcone discretization hidden behind the quantum LBE, Eq. (16). Finally, note that at no point in our treatment did we need to care about stringent symmetry requirements: apparently a simple cubic lattice is good enough to our purpose. This probably relates to the diagonal nature of the quantum-mechanical pressure tensor and to the fact that, unlike fluid dynamics, the theory is *not* self-interacting. Finally, we observe that quantum LBE is as computationally lean and amenable to parallel processing as an explicit scheme can be.

All in all, a good set of credentials for a numerical scheme.

4. Numerical tests

The quantum LBE scheme has been validated on a series of one-dimensional textbook calculations, including

(i) free particle propagation,
(ii) harmonic oscillator,
(iii) scattering from a rectangular barrier [12].

In addition, the scheme has also been demonstrated for simple cases of non-linear Schrödinger equations of direct relevance to Bose–Einstein condensation [13] (as an example, see Fig. 1). These tests provide evidence of the viability of the quantum LBE in one-dimension. The scheme performs efficiently and, what's more, provides *stability* and *unitarity* at a time, a very valuable property for an explicit scheme. As we said, this is related to the peculiar light-cone space-time marching technique inherent to quantum LBE. Higher-dimensional versions akin to the quantum LBE discussed here have been developed systematically by Boghosian and coworkers [8].

5. The quantum N-body problem

In this section we shall explore the question of whether/what the lattice techniques discussed so far can bring any new insight into the problem of solving the Schrödinger equation for a collection of, say, N particles (quantum N-body problem):

$$i\hbar\partial_t \Phi = \sum_{n=1}^{N} \left[-\Delta_n + V(X_n)\right]\Phi, \tag{21}$$

where $X_n = (x_n, y_n, z_n)$ is the spatial coordinate of the nth particle, $\Phi(X_1 \ldots X_N)$ the N-body wavefunction and V the interparticle potential, typically in a two-body format $V(X_n) = \sum_{m>n} V(|X_n - X_m|)$. It has been recently pointed out [9] that quantum lattice algorithms constitute excellent candidates as numerical schemes for quantum computers. In the N-body quantum LBE, each quantum particle is represented by bG walkers, b being the coordination number of the lattice, namely the number of discrete momentum states attached to each lattice site. These walkers move around according to a fictitious microdynamics whose macroscopic limit is precisely the N-body Schrödinger equation.

What would this N-body quantum LBE algorithm look like?

"Simply" evolve N replicas of the single-particle quantum LBE scheme and tie them up together via a two-body potential collecting the sum of all contributions $V_g^n = \sum_{g'} \sum_m V(X_g^n - X_{g'}^m)$ at each given site X_g^n. If one does not insist on the idea of a particle generalized-trajectory, and turns instead to a 'information-network' picture, a generic quantum lattice algorithm would take the form of a first-order, explicit, non-local, map for the complex array Φ_j:

$$\Psi_j(X,t) = \sum_k T_{jk}\Psi_k(X - V_k\Delta t, t - \Delta t), \tag{22}$$

where V_k scans the $3N$-dimensional neighborhood of $X_n = (x_n, y_n, z_n)$, $n = 1, N$ and T_{jk} is the complex transfer matrix fulfilling the unitarity condition $\sum_l T_{jl}T_{lk} = \delta_{jk}$. The kinetic energy operator is sweet since any walker in a given single-particle state can be moved independently of the others, resulting in a linear $O(bGN)$ complexity. Unfortunately, the two-body long-range potential generates a daunting quadratic complexity, $(bGN)^2$, to say nothing of the $(bG)^N$ requirement in computer storage .... The scheme meets

with a "exponential complexity wall" which rules out any possible use of conventional electronic computers for more than a few hundreds particles [17]. Although this statement can probably be challenged by modern multiscale techniques (Achi Brandt, private communication), we shall assume that such exponential complexity is indeed beyond electronic computation capabilities. This brings us back to quantum computers. Since the matter of solving the N-body Schrödinger equation in full on a quantum computer has been described in the existing literature, here we shall take a different path, and discuss how efficient real-space single-particle quantum solvers may contribute to advancing the N-body frontier *without solving the full N-body Schrödinger equation*. Incidentally we note that this is of actual interest not only for current electronic parallel computers, but hopefully also for actual *software emulators of quantum computers* [14,15].

5.1. Quantum LBE and Density Functional Theory

As previously discussed, numerical algorithms for the quantum many-body wavefunction are very hard (to say the least) on electronic computers. Many ways out have been developed to cope with this problem, including,

(i) Quantum Monte Carlo techniques [18],
(ii) Multiscale methods [19],
(iii) Effective one-body theories.

In this paper, we shall be concerned with option (iii).

Effective one-body theories developed in the last fourteen years permit to learn a great deal about the properties of quantum many-body systems without ever invoking the use of many-body wavefunctions. Particularly successful in this respect is the famous Density Functional Theory developed in the 60's by Hohenberg–Kohn and Kohn–Sham [16,17]. The core idea of Density Functional Theory is that the ground state of a many-electron wavefunction (nuclei are regarded as classical particles on account of their higher mass) is uniquely determined by the electronic density $n(\vec{x}) = \sum_j |\phi_j|^2(\vec{x})$, where ϕ_j are one-particle orbitals. The ground-state energy can then be obtained by summing up the single-particle orbital energies obtained by solving the Kohn–Sham equations:

$$H_{\mathrm{KS}}\phi_j = E_j\phi_j, \tag{23}$$

where the Kohn–Sham Hamiltonian consists of four contributions

$$H_{\mathrm{KS}} = -\frac{\hbar^2}{2m}\Delta_j + V_{\mathrm{ext}}(\vec{x}) + \mathrm{e}^2\int \frac{n(\vec{y})}{|\vec{y}-\vec{x}|}\,\mathrm{d}\vec{y} + V_{\mathrm{ex}}[n]. \tag{24}$$

The first two contributions are the usual kinetic energy and external potential operators, the third one relates to the self-consistent Hartree–Fock potential. Finally, the fourth one is an effective 'exchange' energy functional which collects the effects of N-body interactions. The idea is that an effective functional of the electron density exists such that the ground state energy of a fictitious system of *independent* electrons moving in such a potential is *exactly the same* ground-state energy of the interacting system! Describing how such a magic comes about is certainly beyond the scope of this work. Here, we shall simply remark that Density Functional Theory heavily leans on the intuitive picture of a quantum many-body system as a backbone of ions tied up together by a very mobile *electronic fluid*. In this respect, it certainly puts a premium on efficient real-space solvers for the one-particle (non-linear) Schrödinger equation, both in the time-independent (ground-state) and time-dependent (excited states) form. A practical scheme which could be implemented today on either electronic or quantum computer emulators is briefly outlined in the following.

Consider the task of solving a set of N effective time-dependent, one-particle, Kohn–Sham equations coupled via an effective potential $V_{\mathrm{KS}}[\rho]$:

$$\mathrm{i}\hbar\partial_t\phi_j = H_{\mathrm{KS}}\phi_j. \tag{25}$$

Since the LBE grid is uniform, the non-local Hartree–Fock potential is best turned into the corresponding Poisson problem $\Delta V_{\mathrm{HF}} = n$, which is efficiently solved by standard methods such as rapid elliptic solvers or Fast-Fourier techniques. The exchange functional is local and can therefore be handled by the same procedure already discussed and tested for Bose–Einstein condensation.

```
1 |--------- |   |------->
2 |--------- |   |------->
               ====>
3 |--------- |   |------->
4 |--------- |   |------->
  t          t+dt
```

Sketch 1. Parallel solution of the set of N Kohn–Sham equation ($N = 4$). The double line $==$ indicates the serial phase in which each slave processor forwards its partial density to the master and subsequently receives the effective potential to initiate the next step.

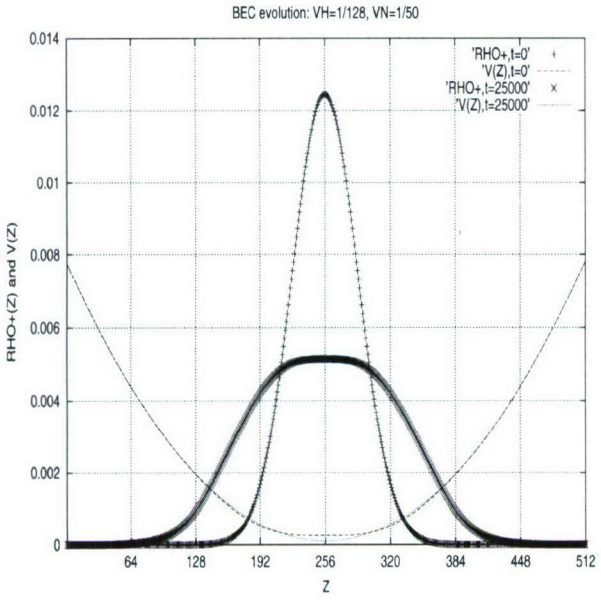

Fig. 1. Probability density $\rho^+(z)$ at $t = 0$ and $t = 25{,}000$ for the case of a self-consistent potential of the form: $V(z) = \frac{4V_H}{L^2}(z - z_0)^2 + V_N\rho$. The flattening of the wavefunction due to self-consistent interactions is well visible. The superscript $+$ indicates the slow, hydrodynamic mode.

The system of Eqs. (25) is also particularly well-suited to parallel computing. In fact, each of these equations can be advanced concurrently in time. Upon completing a single time-step calculation, each processor forwards its partial density $\rho_j(t + \mathrm{d}t) = |\phi_j^2|(t + \mathrm{d}t)$ to a master processor whose task is to collect all contributions, form the effective potential V_{KS} and send it back to each processor to initiate the next time step. This process can be performed fairly efficiently on a electronic parallel since it entails a very lean communication-to-computation ratio. For instance, a parallel computing consisting of $P = N$ processors would solve the N-body problem in (slightly more than) the same time it takes a serial one to solve the single-body quantum equation. Since the same statement applies to a *single-processor* quantum computer, one might dream of quantum-computers applications of paramount scientific problems, such as electronic structure calculations of large molecules of biological interest.

References

[1] O. Madelung, Z. Phys. 20 (1926) 332.
[2] A. Messiah, Quantum Mechanics, North-Holland, 1962.
[3] G. 't Hooft, J. Stat. Phys. 53 (1/2) (1988) 323.
[4] J. Bell, Speakable and Unspeakable in Quantum Mechanics, Cambridge Univ. Press, 1987.
[5] I. Bialynicki-Birula, Phys. Rev. D. 49 (12) (1994) 6920.
[6] S. Succi, R. Benzi, Physica D 69 (1993) 327;
For a review of the Lattice Boltzmann method see: R. Bemzi, S. Succi, M. Vergassola, The Lattice Boltzmann equation: theory and applications, Physics Reports 222 (3) (1992) 145.
[7] J. Wells, V. Oberacker, M. Strayer, A. Umar, in: R. Gruber, M. Tomassini (Eds.), Proceedings 6th Int. Conf. on Physics Computing, European Phys. Society, 1994, p. 655.
[8] B. Boghosian, W. Taylor, Int. J. Mod. Phys. C 8 (4) (1997) 705.
[9] D. Meyer, Int. J. Mod. Phys. C 8 (4) (1997) 717.
[10] R. Feynman, R. Feynman, A. Hibbs, Quantum Mechanics and Path Integrals, McGraw-Hill, New York, 1965.
[11] K. Kulander (Ed.), Comp. Phys. Commun. 63 (1991), Special issue on Time-dependent methods for quantum dynamics.
[12] S. Succi, Phys. Rev. E 53 (2) (1996) 1969.
[13] S. Succi, Int. J. Mod. Phys. C 9 (8) (1998) 1577.
[14] D.P. DiVincenzo, Science 270 (1995) 255.
[15] J. Yepez, Int. J. Mod. Phys. C 9 (8) (1998) 1587.
[16] P. Hohenberg, W. Kohn, Phys. Rev. 136 (1964) B864.
[17] W. Kohn, L. Sham, Phys. Rev. 140 (1965) A1133.
[18] D. Ceperley, B. Alder, Phys. Rev. Lett. 45 (1980) 566.
[19] T. Beck, Rev. Mod. Phys. 72 (4) (2000) 1041.

ELSEVIER

Computer Physics Communications 146 (2002) 324–330

Computer Physics Communications

www.elsevier.com/locate/cpc

A quantum full adder for a scalable nuclear spin quantum computer

G.P. Berman [a,*], G.D. Doolen [a], G.V. López [b], V.I. Tsifrinovich [c]

[a] *T-13 and CNLS, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*
[b] *Departamento de Física, Universidad de Guadalajara Corregidora 500, S.R. 44420, Guadalajara, Jalisco, Mexico*
[c] *IDS Department, Polytechnic University Six Metrotech Center, Brooklyn, NY 11201, USA*

Received 14 February 2001

Abstract

We demonstrate a strategy for implementation a quantum full adder in a spin chain quantum computer. As an example, we simulate a quantum full adder in a chain containing 201 spins. Our simulations also demonstrate how one can minimize errors generated by non-resonant effects. © 2002 Published by Elsevier Science B.V.

PACS: 03.67.Lx; 03.67.-a; 76.60.-k

Keywords: Qubit; Full adder; Quantum computer

1. Introduction

A full adder is a basic component of a conventional computer and a welcome asset for quantum computers. In particular, the Shor quantum algorithm requires modular exponentiation, $f(x) = a^x \pmod N$, which cannot be computed without a quantum full adder. The question arises: What is a full adder in a quantum computer? A quantum computer operates on a superposition of numbers simultaneously. What is not clear is what it means to add two superpositions. One definition of a full adder in a quantum computer is that a full adder is a gate which adds a given number to a superposition of numbers. This full adder must simultaneously add a definite number, "a", to all numbers, "b_i", which are in a superposition in a quantum computer register. The basic idea for the full adder is well known [1,2]. However it is not clear how to implement this idea in a many-qubit spin quantum computer. It is even less clear how to simulate accurately a quantum full adder having a large number of qubits using a conventional computer. One must understand the role of non-resonant effects and how to minimize them. One must also know the structure of the error states caused by non-resonant effects. Our paper provides an answer for these issues. To implement a quantum full adder, we propose to use a sequence of electromagnetic π-pulses on a spin-chain quantum computer. We have simulated the dynamics of this quantum full adder with 201 spins, on a conventional computer; analyzed unwanted non-resonant effects; and determined the structure of the error states and ways to reduce non-resonant effects.

* Corresponding author.
E-mail address: gpb@lanl.gov (G.P. Berman).

0010-4655/02/$ – see front matter
PII: S0010-4655(02)00423-X

Table 1
Table of binary addition

a	b	c	s'	c'	$ab \oplus ac \oplus bc$
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	1	0	0
1	0	0	1	0	0
0	1	1	0	1	1
1	0	1	0	1	1
1	1	0	0	1	1
1	1	1	1	1	1

2. A classical full adder

A classical full adder operates with an input of two addend bits, "a" and "b", and a carry bit, "c". (See Table 1.) In Table 1, s' and c' are the output sum and the carry-over, respectively. The sum, s', can be easily expressed as $s' = a \oplus b \oplus c$ (where $\oplus$ is addition modulo 2). The formula for the carry-over is more complicated. One can see from Table 1 that $c' = ab \oplus ac \oplus bc$.

3. A quantum full adder

We describe the basic idea of a quantum full adder first suggested in [1]. The full adder quantum gate (F) depends on the value of the bit "a". If $a = 0$, the quantum computer applies the gate $F(0)$. If $a = 1$, it applies $F(1)$. The F-gate can be expressed in terms of the Control-Not (C_{ik}) and Control–Control-Not (C_{ikp}) gates. A C_{ik}-gate changes the value of the target qubit, "k", if the control qubit, "i", has the value "1". A C_{ikp}-gate changes the state of the target qubit, "p", if both qubits, "i" and "k", have the value "1",

$$\begin{aligned} &C_{ik}|\ldots n_i \ldots n_k \ldots\rangle = \big|\ldots n_i \ldots (n_i \oplus n_k)_k \ldots\big\rangle, \\ &C_{ikp}|\ldots n_i \ldots n_k \ldots n_p \ldots\rangle \\ &\quad = \big|\ldots n_i \ldots n_k \ldots \big[(n_i n_k) \oplus n_p\big]_p \ldots\big\rangle. \end{aligned} \tag{1}$$

We shall also use the Not-gate which can be designated by C_i. It changes the value of the ith qubit independent of the values of all other qubits. Using these gates, we can transform the state, $|0_3 c_2 b_1\rangle$, into the state, $|c'_3 s'_2 b_1\rangle$,

$$F_{321}(a)|0_3 c_2 b_1\rangle = \big|c'_3 s'_2 b_1\big\rangle, \tag{2a}$$

where "a" and "b" are the addend bits and s' is their sum, c and c' are the input and output carry bits. Namely,

$$\begin{aligned} &F_{321}(0) = C_{12} C_{123}, \\ &F_{321}(1) = C_{12} C_{123} C_2 C_{23}. \end{aligned} \tag{2b}$$

We use the convention that the right gate acts first. Let us check the action of the full adder, F_{321}, for example, for $a = b = c = 1$. We use $F_{321}(1)$:

$$C_{23}|0_3 c_2 b_1\rangle = C_{23}|0_3 1_2 1_1\rangle = |1_3 1_2 1_1\rangle. \tag{3}$$

In (3), the second control qubit has the value "1". So, the third target qubit changes its value from "0" to "1". Next,

$$\begin{aligned} &C_2|1_3 1_2 1_1\rangle = |1_3 0_2 1_1\rangle, \\ &C_{123}|1_3 0_2 1_1\rangle = |1_3 0_2 1_1\rangle, \\ &C_{12}|1_3 0_2 1_1\rangle = |1_3 1_2 1_1\rangle = \big|c'_3 s'_2 b_1\big\rangle. \end{aligned} \tag{4}$$

Thus, in accordance with the Table 1, we obtain,

$$b = 1, \quad s' = 1, \quad c' = 1.$$

4. A spin chain quantum computer

An atomic chain quantum computer based on triplets, $ABC\,ABC\,ABC\ldots$, was first introduced in [3]. The implementation of this idea for a chain of spins interacting through the Ising interaction was given in [4]. Ising-type spin chains have been used for quantum computation in a statistical ensemble of quantum computers [5,6]. In [7], we considered a simple model—a chain of identical spins in a non-uniform magnetic field.

The present paper is based on [4,7]. We consider a chain of spins (e.g., nuclear spins) in a non-uniform magnetic field. Similar to [7], the angle, Θ, between the direction of the chain and the direction of the permanent magnetic field (z-direction) satisfies the condition: $\cos\Theta = 1/\sqrt{3}$. Then, the dipole–dipole interaction between spins is suppressed, and the Ising interaction becomes dominant. We suppose that our chain consists of a periodic sequence of triplets, $ABC\,ABC\,ABC\ldots$. The triplet, ABC, can be different nuclear spins or identical spins in slightly different environments. For definiteness, we shall keep in mind

this second case [7] with the following typical parameters,

$$\omega_1/2\pi \sim 400\ \text{MHz}, \quad \omega_k = \omega_1 + (k-1)\Delta\omega,$$
$$\Delta\omega/2\pi \approx 20\ \text{kHz}, \tag{5}$$
$$J_{AC}/2\pi \approx 100\ \text{Hz}, \quad J_{BC} = 2J_{AC}, \quad J_{AB} = 3J_{AC},$$

where ω_1 is the Larmor (angular) frequency of the right end spin, ω_k is the Larmor frequency of the kth spin, J_{ik} is the constant of Ising interaction between the ith and kth spins.

In the presence of a circularly polarized transverse magnetic field, the Hamiltonian of the system can be written as [8],

$$\mathcal{H} = -\sum_k \omega_k I_k^z - 2\sum_k J_{k,k+1} I_k^z I_{k+1}^z - \frac{\Omega}{2}\sum_k \{I_k^- \exp[-\mathrm{i}(\omega t+\varphi)] + I_k^+ \exp[\mathrm{i}(\omega t+\varphi)]\}, \tag{6}$$

where I^z is the nuclear spin operator, ω and φ are the frequency and the phase of the transverse magnetic field; Ω is the Rabi frequency (the magnitude of the transverse magnetic field in frequency units), and $\hbar = 1$.

5. Implementation of the quantum full adder

First we shall define our problem. Suppose that we have a number $|b\rangle = |b^{(L-1)}\ldots b^{(0)}\rangle$, $b^{(m)} = 0, 1$. In decimal notation, $b = b^{(L-1)}2^{L-1} + \cdots + b^{(0)}2^0$. (Below, we omit parentheses in the superscripts.) Note, that in general, we have a superposition of many numbers, $|b\rangle_i$, and any gate must act on all of these numbers simultaneously. We are going to add to a number $|b\rangle$ (all numbers, b_i, in a superposition), a definite number, $a = a^{L-1}\ldots a^0$, where $a^m = 0, 1$. To achieve this goal we shall use $2L+1$ qubits. We load the number b in a chain of qubits in the following way,

$$|0_{2L+1} b_{2L}^{L-1} 0_{2L-1} b_{2L-2}^{L-2} \ldots 0_5 b_4^1 0_3 0_2 b_1^0\rangle. \tag{7}$$

This means that we place two additional qubits in the states $|0\rangle$ in front of the qubit b_1^0, and one additional qubit in the state $|0\rangle$ in front of all other qubits b^m, $m \neq 0$.

The gate C_{123} is not convenient for our spin chain quantum computer in which each spin interacts only with its neighbors. So, we replace it with the following sequence of gates,

$$C_{123} = C_{23}C_{32}C_{23}C_{132}C_{23}C_{32}C_{23}. \tag{8}$$

Let us check this equation, for example, for the state, $|0_3 1_2 1_1\rangle$. From the left side of Eq. (8) we have,

$$C_{123}|0_3 1_2 1_1\rangle = |1_3 1_2 1_1\rangle, \tag{9}$$

as the first and the second qubits are in the states, $|1\rangle$, the third qubit changes its state. Next, we follow the operations on the right-hand side of (8) to show that the same result is obtained. From the right side of Eq. (8) we have,

$$\begin{aligned} &C_{23}|0_3 1_2 1_1\rangle = |1_3 1_2 1_1\rangle, \quad C_{32}|1_3 1_2 1_1\rangle = |1_3 0_2 1_1\rangle,\\ &C_{23}|1_3 0_2 1_1\rangle = |1_3 0_2 1_1\rangle, \quad C_{132}|1_3 0_2 1_1\rangle = |1_3 1_2 1_1\rangle,\\ &C_{23}|1_3 1_2 1_1\rangle = |0_3 1_2 1_1\rangle, \quad C_{32}|0_3 1_2 1_1\rangle = |0_3 1_2 1_1\rangle,\\ &C_{23}|0_3 1_2 1_1\rangle = |1_3 1_2 1_1\rangle, \end{aligned} \tag{10}$$

which coincides with the right side of Eq. (9). Now, instead of (2b), we have the following expression for the full adder, F,

$$\begin{aligned} F_{321}(0) &= C_{12}C_{23}C_{32}C_{23}C_{132}C_{23}C_{32}C_{23},\\ F_{321}(1) &= C_{12}C_{23}C_{32}C_{23}C_{132}C_{23}C_{32}C_{23}C_2C_{23}. \end{aligned} \tag{11}$$

Next, we explain how to add the numbers b and a. If $a^0 = 0$, the quantum computer applies the gate, $F_{321}(0)$. If $a^0 = 1$, the quantum computer applies the gate, $F_{321}(1)$. According to (2), the result is,

$$F_{321}|0_3 0_2 b_1^0\rangle = |c_3^1 s_2^0 b_1^0\rangle, \tag{12}$$

where we have replaced s' by $s^0 = a^0 \oplus b^0$, and c' by c^1 which is the carry-over for the next addition of b^1 and a^1.

Next, consider the five right most qubits,

$$|0_5 b_4^1 c_3^1 s_2^0 b_1^0\rangle. \tag{13}$$

To add b^1 and a^1 and the carry-over, c^1, we should first make a swap, S, of the values of the fourth and the third qubits,

$$S_{43}|b_4^1 c_3^1\rangle = |c_4^1 b_3^1\rangle. \tag{14}$$

The swap gate, S_{ik}, can be represented in terms of C_{ik} gates,

$$S_{ik} = C_{ki}C_{ik}C_{ki}. \tag{15}$$

Let us check, for example, the action of the S_{21} gate on the state, $|0_2 1_1\rangle$. Using (15), we have,

$$S_{21} = C_{12}C_{21}C_{12},$$
$$C_{12}|0_2 1_1\rangle = |1_2 1_1\rangle, \quad C_{21}|1_2 1_1\rangle = |1_2 0_1\rangle, \quad C_{12}|1_2 0_1\rangle = |1_2 0_1\rangle. \tag{16}$$

Thus, the S-gate transforms the state, $|01\rangle$, into the state, $|10\rangle$.

After the action of the S_{43} gate, the state (13) changes into,

$$|0_5 c_4^1 b_3^1 s_2^0 b_1^0\rangle. \tag{17}$$

The state, $|0_5 c_4^1 b_3^1\rangle$, has the form (2a), and it is ready for application of the full adder, F_{543},

$$F_{543}|0_5 c_4^1 b_3^1\rangle = |c_5^2 s_4^1 b_3^1\rangle. \tag{18}$$

Certainly, we use $F_{543}(0)$ if $a^1 = 0$ and $F_{543}(1)$ if $a^1 = 1$. Thus, we obtain the sum, $s^1 = a^1 \oplus b^1 \oplus c^1$, and a carry-over, c^2, for the next addition. It is clear that by repeating the application of gates F and S, we shall obtain the desired answer. A complete full adder, F, can be represented as a combination of the elementary full adders, F_{ijk}, and the swap-gates, S_{ij},

$$F = F_{2L+1,2L,2L-1}(a^{L-1}) S_{2L,2L-1} \cdots F_{765}(a^2) S_{65} F_{543}(a^1) S_{43} F_{321}(a^0). \tag{19}$$

After the action of the full adder, the superposition of the states (7) transforms into the superposition of states,

$$|c^L s^{L-1} b^{L-1} s^{L-2} b^{L-2} \ldots s^0 b^0\rangle. \tag{20}$$

Thus, the qubits in even positions and the left end qubit carry the results of addition. The problem is: How to implement all of these gates using π-pulses? We suppose that $\Omega \ll J_{ik}$, say $\Omega/2\pi \approx 10$ Hz. This allows us to excite each spin individually. To implement F_{321}, we use Eqs. (11). As an example, we show how to implement $F_{321}(0)$. According to Eqs. (11), first we implement C_{23}, then C_{32}, and so on. Suppose we have an integer number of triples, ABC, in our spin chain. Then, we have 4 possible frequencies in the position $(3n-2)$ which correspond to a spin C:

$$\begin{aligned} \omega_{3n-2}^{00} &= \omega_{3n-2} + J_{BC} + J_{AC}, \\ \omega_{3n-2}^{01} &= \omega_{3n-2} + J_{BC} - J_{AC}, \\ \omega_{3n-2}^{10} &= \omega_{3n-2} - J_{BC} + J_{AC}, \\ \omega_{3n-2}^{11} &= \omega_{3n-2} - J_{BC} - J_{AC}. \end{aligned} \tag{21}$$

Here ω_k^{ij} corresponds to the kth spin whose left neighbor is in state $|i\rangle$ and whose the right neighbor is in state $|j\rangle$. Similar expressions can be found for spins in positions $(3n-1)$ and $(3n)$, the B and A spins, respectively. The end spins have only two frequencies. For the left end spin, A,

$$\omega_{2L+1}^0 = \omega_{2L+1} + J_{AB}, \quad \omega_{2L+1}^1 = \omega_{2L+1} - J_{AB}, \tag{22}$$

and for the right end spin, C,

$$\omega_1^0 = \omega_1 + J_{BC}, \quad \omega_1^1 = \omega_1 - J_{BC}. \tag{23}$$

Now, to implement C_{23}, we apply a π-pulse with frequency ω_3^{01} and then a π-pulse with the frequency ω_3^{11}. One of these two pulses definitely changes the state of the third spin if the second spin is in its excited state, $|1\rangle$. To implement the gate C_{123}, we need a single π-pulse with frequency ω_2^{11}. To implement a Not-gate, C_2, which appears in $F_{132}(1)$, we have to apply four π-pulses with all possible frequencies, ω_2^{ij}, $i = 0, 1$, and $j = 0, 1$. The total number of pulses required to implement a $F_{ijk}(0)$-gate (if $i \neq 2L+1$, $k \neq 1$) is 15: two pulses for each Control-Not gate and one pulse for the Control–Control-Not gate. For the $F_{ijk}(1)$ gate, the total number of pulses is 21 (16 pulses for the left triple of the chain). The swap gate, S_{ij}, requires 6 pulses.

Thus, to add L-qubit numbers, our proposal requires $(2L+1)$ qubits and less than $27L$ π-pulses.

6. Simulations of a quantum full adder

For numerical simulations of a quantum full adder we used the following assumption: the frequency difference between two neighboring spins is much greater than the Rabi frequency. As a result, the selective excitation of a chosen spin has a small effect on all other spins. At the same time, we take into account the action of a π-pulse on non-resonant states. For example, suppose that the frequency of a π-pulse is: $\omega = \omega_3^{11}$. This pulse is resonant with spin "3" only if the neighboring spins, "1" and "2", are in their excited states. The states in which the spin "3" has frequencies, ω_3^{10}, ω_3^{01}, and ω_3^{00}, are non-resonant for the pulse with $\omega = \omega_3^{11}$. We take into consideration the transformations of all these non-resonant states.

Note, that any pulse in our simulations acts on all basic states in the quantum superposition. The resonant state transforms into a state with the opposite direction of the resonant spin. Every non-resonant state transforms into the superposition of two states: the initial one and an error state generated by the non-resonant transition. If the probability of an error state is less than a chosen small number, ε, our computer program automatically removes it. One can argue that every removed state can generate a number of new states. However, suppose that we have removed a state with a small probability, P. Due to the main property of the unitary transformations, the sum of the probabilities of all states generated by the removed state (including itself) is P. Thus, the total probability of all neglected states cannot increase in spite of generation of new states!

We have simulated the addition of the 100-digit numbers in a 201 spin chain. To minimize non-resonant errors, we used the $2\pi k$-method [7,8]. The basic idea of the $2\pi k$-method is the following. One chooses the Rabi frequency, Ω, of a resonant π-pulse in such a way that it becomes approximately a $2\pi k$-pulse for all non-resonant transitions (where k is an integer which generally is different for different states). A $2\pi k$-pulse does not generate unwanted error states. A transformation of a basic state under the action of a π-pulse was described in our previous paper [7]. The following values of dimensionless parameters were chosen,

$$J_{AC} = 1, \quad J_{BC} = 2, \quad J_{AB} = 3. \tag{24}$$

We found that all non-resonant transitions approximately satisfied the $2\pi k$-conditions for the value of the Rabi frequency, $\Omega = \Omega_0 = 0.10005$.

Next, we present the results of our numerical simulations. As an example, we add the "classical number",

$$1^{99}0^{98}\ldots0^{0} \tag{25}$$

(2^{99} in a decimal notation) to the "quantum number",

$$0^{99}\ldots0^{1}1^{0} \tag{26}$$

(1 in a decimal notation). The sum of these two numbers,

$$1^{99}0^{98}\ldots0^{1}1^{0}, \tag{27}$$

corresponds to the quantum state of the chain of 201 spins,

$$|0_{201}1_{200}0_{199}\ldots0_{3}1_{2}1_{1}\rangle. \tag{28}$$

For the value of $\Omega = \Omega_0$ which approximately satisfies $2\pi k$-condition, we have for the probability of the expected state (28): $P_0 = 0.99889$. The number, N, of error states with probability $P > 10^{-12}$ is only 304. Small deviations from the $2\pi k$-condition significantly influence the result. As an example, for $\Omega = 0.10021$, the value of P_0 decreases to 0.98300, and the number of error states, N, grows to 46530. Fig. 1 shows the dependence of P_0 and N on the Rabi frequency, Ω in the vicinity of Ω_0. One can see that the probability, P_0, smoothly decreases by approximately 1% when the Rabi frequency, Ω, shifts from Ω_0 by approximately 0.1%. Unlike the value of P_0, the number of error states with probability $> 10^{-12}$ does not change in the close vicinity of Ω_0. But it sharply increases when the deviation of Ω from Ω_0 approaches 0.15%. Fig. 2 shows the probabilities

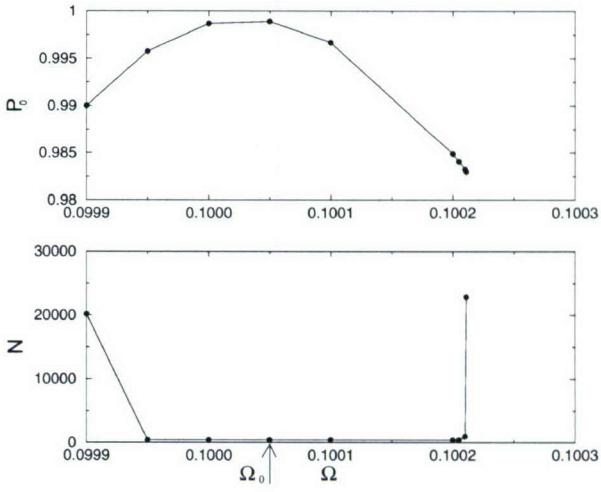

Fig. 1. The probability of the expected state, P_0; and the number of error states, N, as a function of the Rabi frequency, Ω.

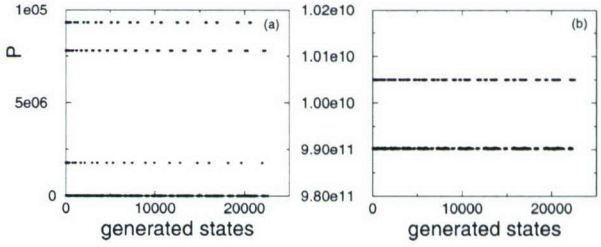

Fig. 2. "Line spectrum" of the probabilities, P, of error states ($\Omega = 0.10021$); (a) the region: $P \sim 10^{-6}$, (b) the region: $P \sim 10^{-10}$.

of error states (the states are shown in the order of their generation). One can see a specific "line spectrum" of the probabilities: the error states are "attracted" to a few discrete values of the probability. Very similar effects were obtained for other examples of addition including the addition of a "classical" number to a superposition of two quantum numbers.

7. Phase control

The quantum full adder implementation considered above provides a transformation of an arbitrary superposition of "quantum numbers", q_k, into a superposition of numbers $(q_k + a)$, where a is any given "classical number". However, this proposed scheme also generates complicated phase differences between the states of the quantum superposition. This effect can be inappropriate, especially for the Shor algorithm. In this section, we describe how to extend our simple model to incorporate phase restoration after the action of every π-pulse.

Consider the chain of paramagnetic ions ABCABC containing nuclear spins in a high nonuniform magnetic field. Every electron spin interacts with the nuclear spin of its ion via the hyperfine interaction. Electron spins interact with each other through the Ising interaction. All electron spins are in their ground states. The nuclear spins also interact with each other through the Ising interaction. This interaction is responsible for quantum logic gates. The dipole–dipole interaction is suppressed due to the use of the magic angle between the chain and the direction of the external magnetic field. The key point of our model is the following: we consider the interaction (for simplicity, the Ising interaction) between an electron spin and its two neighboring nuclear spins. This interaction can originate, for example, if the electron density at the neighboring nuclei is not zero. Thus, the electron spin frequency for a particular ion can take eight values. As an example, for the B-ion one has,

$$\begin{aligned}
\omega^{000} &= \omega_e + J_{ee}^{AB} + J_{ee}^{BC} + A^B/2 + J_{en}^{AB} + J_{en}^{BC},\\
\omega^{100} &= \omega_e + J_{ee}^{AB} + J_{ee}^{BC} - A^B/2 + J_{en}^{AB} + J_{en}^{BC},\\
\omega^{110} &= \omega_e + J_{ee}^{AB} + J_{ee}^{BC} - A^B/2 - J_{en}^{AB} - J_{en}^{BC},
\end{aligned} \tag{29}$$

and so on. Here ω^{ijk} corresponds to the electron frequency for the case in which the nuclear spin of the same ion is in the state $|i\rangle$, the nuclear spin of the neighboring A-ion is in the state $|k\rangle$, and the nuclear spin of the neighboring C-ion is in the state $|j\rangle$; ω_e is the Zeeman frequency, J_{ee}^{AB} and J_{ee}^{BC} are the constants of the electron–electron interaction, A^B is the hyperfine constant (in the frequency units), and J_{en}^{AB} and J_{en}^{BC} are the constants of the electron–nuclear interaction for neighboring ions.

Thus, the ESR frequency depends on the position of an electron spin in the chain (because the Zeeman frequency is nonuniform) and the states of three nuclear spins—the nuclear spin of its own ion (via the hyperfine interaction) and the nuclear spins of two neighboring ions. One can tune the frequency of an electromagnetic pulse in such a way that it is resonant with the electron spin only if it is in a definite position in the chain and the three nuclear spins mentioned above are in definite states.

The strategy for phase correction for the quantum full adder is the following. The full adder is implemented by a sequence of the nuclear π-pulses. A nuclear π-pulse causes a phase shift of $\pi/2$ for the resonant states. There are six possible phase distortions for non-resonant states (3 possible states for neighboring nuclear spins times two possible states for the selected nuclear spin). Correspondingly, one has six possible frequencies for the electron spin of the selected ion in the non-resonant state. Because of the large value of the electron gyromagnetic ratio, we assume that each of the corresponding electron transitions can be driven without noticeable non-resonant effects on the electron transitions with close frequencies. After the nuclear π-pulse, one applies 12 electron π-pulses: two π-pulses for every possible frequency of the selected electron spin in the non-resonant state. The phase of the first π-pulse is zero, the phase of the second one is ϕ. The total phase shift of the wave function of the ion is $(\pi + \phi)$. By choosing an appropriate value for ϕ, one can change the phase shift for a specific non-resonant state to the value of $\pi/2$, which corresponds to the resonant state. After the action of 12 electron π-pulses, all states in the quantum superposition will have the same phase, and all electron spins will be in their ground state. Thus, the phase distortion generated by a nuclear π-pulse can be corrected with 12 electron π-pulses.

8. Conclusions

We have demonstrated a strategy for implementation of a full adder in a nuclear spin quantum computer. To add an arbitrary superposition of L-qubit "quantum numbers" and a fixed L-bit "classical number", our scheme requires $2L + 1$ qubits and less than $27L$ resonant π-pulses. We have simulated the action of this full adder for a spin chain containing 201 qubits. Using the $2\pi k$-method allowed us to minimize the generation of unwanted non-resonant error states. These error states are shown to have a few preferred discrete values of probability (a "line spectrum"). Our simple spin model is generalized to correct undesired phase distortions.

Acknowledgements

The work was supported by the Department of Energy (DOE) under contract W-7405-ENG-36, by the National Security Agency (NSA) and by the Advanced Research and Development Activity (ARDA).

References

[1] D. Beckman, A.N. Chari, S. Devabhaktuni, J. Preskill, Phys. Rev. A 54 (1996) 1034.
[2] A.O. Pittenger, An Introduction to Quantum Computing Algorithms, Birkhäuser, 1999.
[3] S. Lloyd, Science 261 (1993) 1569.
[4] G.P. Berman, G.D. Doolen, D.D. Holm, V.I. Tsifrinovich, Phys. Lett. A 193 (1994) 444.
[5] N.A. Gershenfeld, I.L. Chuang, Science 275 (1997) 350.
[6] D.G. Cory, A.F. Fahmy, T.F. Havel, Proc. Natl. Acad. Sci. USA 94 (1997) 1634.
[7] G.P. Berman, G.D. Doolen, G.V. López, V.I. Tsifrinovich, Phys. Rev. A 61 (2000) 2305.
[8] G.P. Berman, G.D. Doolen, R. Mainieri, V.I. Tsifrinovich, Introduction to Quantum Computers, World Scientific, 1998.

ELSEVIER

Computer Physics Communications 146 (2002) 331–338

Computer Physics Communications

www.elsevier.com/locate/cpc

Quantum computing with spin qubits in semiconductor structures

Vladimir Privman *, Dima Mozyrsky, Israel D. Vagner

Department of Physics and Center for Advanced Materials Processing, Clarkson University, Potsdam, NY 13699-5820, USA

Received 16 February 2001

Abstract

We survey recent work on designing and evaluating quantum computing implementations based on nuclear or bound-electron spins in semiconductor heterostructures at low temperatures and in high magnetic fields. General overview is followed by a summary of results of our theoretical calculations of decoherence time scales and spin–spin interactions. The latter were carried out for systems for which the two-dimensional electron gas provides the dominant carrier for spin dynamics via exchange of spin-excitons in the integer quantum Hall regime.

PACS: 73.20.Dx; 71.70.Ej; 03.67.Lx; 76.60.-k

Keywords: Quantum computing; Semiconductor; Decoherence; Spin; Qubit; Electron gas

1. Introduction

The field of quantum computing has seen explosive growth of experimental and theoretical interest. The promise of quantum computing [1–5] has been in exponential speedup of certain calculations via quantum parallelism. In Fig. 1, the top flow chart shows the "classical" computation which starts from binary input states and results in binary output states. The actual dynamics is not really that of Newtonian classical mechanics. Rather the computation involves many-body irreversible "gate" device components, made of semiconductor materials in modern computers, which evolve irreversibly, "thermodynamically" according to the laws of statistical mechanics. As the size of the modern computer components approaches atomic, the many-body quantum behavior will have to be accounted for in any case [6].

The idea of quantum computing, however, is not just to account for, but to actually utilize the quantum-mechanical dynamical behavior. This is not an easy task. Quantum mechanics allows for parallelism in evolution: one can "process" a linear superposition of several input states at once, as illustrated in the lower flow chart in Fig. 1. The price paid is that coherent processing of information, according to the law of quantum mechanics, must be accomplished in systems much larger than atomic-size (or more importantly, with many degrees of freedom). There are numerous conceptual and experimental obstacles to accomplishing this task, that have generated a lot of interest, excitement, and new results in computer science, physics, and engineering.

The functioning of a quantum computer involves initialization of the input state, then the actual dynamical evolution corresponding to computation, and finally reading off the result. Various specific requirements for implementation have been identified

* Corresponding author.
E-mail address: privman@clarkson.edu (V. Privman).

0010-4655/02/$ – see front matter
PII: S0010-4655(02)00424-1

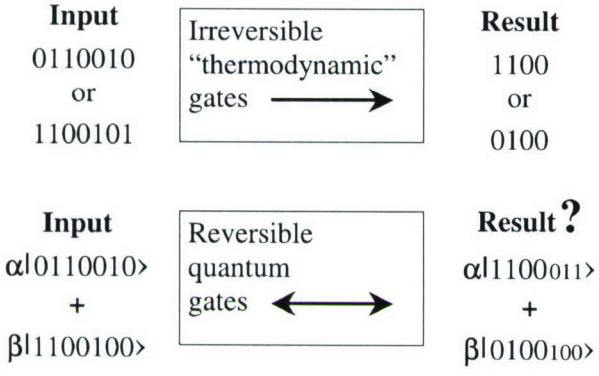

Fig. 1. Comparison of the classical and quantum approaches to computing. The upper flow chart schematically represents implementation of a traditional irreversible "classical" computation process, where transformation of the input set of bits into the result is accomplished by a succession of irreversible gates. Owing to their irreversibility, the gates can be connected in space rather than switched on and off at different times. The lower flow chart shows quantum processing of information, where the input and the final result are both in superposition states, yielding quantum parallelism. The dynamics is reversible: there is a one-to-one correspondence between the initial and final states. Therefore, number of the input and output quantum bit (qubits) is the same even though some of the output qubits (set in a smaller font) might not be used in the final extraction of the classical result by measurement. The quantum gates are applied in succession by being switched on and of at different times during the computation. The question mark signifies the difficulty of finding quantum algorithms that retain the power of quantum parallelism after measurement needed to read off the final result as classical information.

[2–5]; here we provide only a limited introductory overview.

Let us begin by considering the reading off of the final result. The reason for the question mark in the lower chart in Fig. 1 is that quantum measurement of the final superposition state can erase the gain of the parallel dynamics, by collapsing the wave function. Therefore, a key issue in quantum computing has been to find those algorithms for which the readout of the final state, by way of projecting out a certain average property, still retains the power of the quantum parallelism. To date, only few such examples are known [1,3,4,7], the most celebrated being the Shor algorithm [1] for factoring of integers, the invention of which boosted quantum computing from an obscure theoretical field to a mainstream research topic.

The preparation of the initial state does not seem to present a problem for most quantum computing realizations [2–5], except perhaps the ensemble liquid state NMR approach [8,9] which relies on the initial thermal distribution to produce deviation of the density matrix from the equal-probability mixture state. In most other approaches, the initial state can be produced by first fully polarizing the quantum bits (qubits), i.e. putting them in one of the two quantum levels. Note that we consider two-state qubits here, realized, for instance, by spins 1/2 of nuclei or gate- or impurity-bound electrons, in applied magnetic field. The fully polarized state is then subject to gate operations to form the desired input state. Part of a quantum-computing algorithm should be the prescription on how to choose the initial state to represent the classical information of the input, like the input integer in the factoring. In most cases, this prescription is easily accomplished by single-qubit and two-qubit gates.

The actual dynamical evolution (the process of computation) in quantum computing is fully reversible and nondissipative, unlike classical computing. Much progress has been made in resolving both the conceptual and computer-engineering "design" issues for quantum computation. Specifically, the computation can be carried out [2–5,10–13] by a universal set of gates: single-qubit rotations and nearly any two-qubit gate. The gates are not connected in space like in classical computers but are activated in succession in time, to control single-spin dynamics and also switch on and off two-spin interactions (we use "spin" and "qubit" interchangeably).

Many interesting matters have been resolved, which are not reviewed here. These include the understanding of how the finiteness of the state space (i.e. two states for spin 1/2) replaces the "classical" digitalization in quantum computing. Also, the "classical" copying (fan-out) function is not possible in quantum mechanics. It is replaced by entanglement with ancillary qubits to accomplish redundancy needed for error correction [14–20]. Sources of errors due to interactions with environment in quantum mechanics involve not only the usual relaxation (thermalization) but also loss of coherence [21–28]. This quantum decoherence (dephasing) can be faster than relaxation because it does not require energy exchange.

A conceptually important issue has been the scalability of quantum computing: can one process macroscopically large amounts of information by utilizing quantum error correction based on redundancy via entanglement with ancillary qubits? The affirmative an-

swer to this question has been one of the triumphs of the theory [14–20]. It provided a new paradigm for emergence of controlled/organized macroscopic behavior from microscopic dynamics, on par with the conceptual possibility of living organisms, which we observe but cannot yet "manufacture", and million-gate classical computers which are man-made.

With all these theoretical advances at hand, the next step is to ask whether a man-made quantum computer can be realized? There have been several experimental directions of exploration, most presently are still at the level of one or two qubits, or, for ensemble liquid-state NMR, which emulates quantum dynamics by evolution of the density matrix of a large collection of molecules, 5–7 qubits.

In this introductory survey, we summarize results of our work on two-spin interactions and spin decoherence in semiconductor heterostructures. In Section 2, we consider the spin-based quantum computing proposals in such systems. Time scales of relaxation and decoherence are addressed in Section 3. Finally, Section 4 reports results for models with nuclear spins as qubits.

2. Spin-based quantum computing in semiconductor heterostructures

The general layout of a solid-state quantum computer is shown in Fig. 2. Qubits are positioned with precision of few nanometers in a heterostructure. One must propose how to effect and control single-qubit interactions, two-qubit interactions, and explore how the controlled dynamics owing to these interactions compares to decoherence and relaxation. The proposal must include ideas for implementation of initialization, readout, and gate functions.

The first proposal including all these components was for qubits realized in an array of quantum dots [29] coupled by electron tunneling. The first spin-based proposal [30] utilized nuclear spins coupled by the two-dimensional electron gas, the latter in the dissipationless integer quantum Hall state [31] that requires low temperatures and high magnetic fields. An important advancement was the work of Kane [32] where gate control of nuclear-spins of donor impurities, separated less than 10 nm and coupled via the outer impurity electrons which are bound at low temperatures, was proposed. Most of these ideas also apply to electron-spin qubits, bound at impurities, in quantum dots, or directly by gates. Several elaborate solid-state heterostructure quantum computing schemes have been proposed in the literature recently [28,33–41]. There are also other promising proposals involving surface geometries: superconducting electronics [42–46] and electrons on the surface of liquid helium [47].

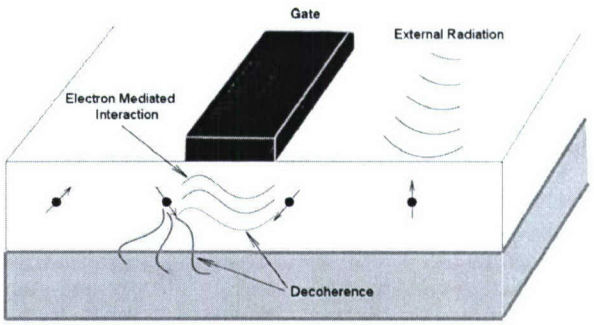

Fig. 2. Schematic illustration of a semiconductor heterostructure quantum information processor. The qubits, represented by the arrows overlaying heavy dots, are spins 1/2 of nuclei or localized electrons. Individual control of the temporal evolution of the spins can be achieved with the use of external electromagnetic radiation, i.e. NMR or ESR pulses. The spins are also coupled with each other via interaction mediated by the two-dimensional electron gas in the heterostructure, or by other means. The external and internal interactions can be controlled by gates formed on top of the heterostructure. The external environment, that includes crystal lattice, electron gas, defects, impurity potentials, causes relaxation and decoherence of the qubits.

There have been several planned and ongoing experimental efforts [32–36,43–45,48–54] ultimately aimed at solid-state quantum computing and other quantum information processing realizations. The final geometry is expected to be most sensitive to the implementation of readout, because it involves quantum measurement, i.e. supposedly interaction with or transfer of information to a macroscopic device. Therefore, much of the experimental effort presently has been focused on single-qubit (single-spin) measurement approaches.

The theoretical efforts can be divided into two majors tasks. The process of single-spin measurement must be understood for the readout stage of quantum computing. Several conceptual and calculational advancements have been made in understanding quantum measurement [26,32–36,46,50,55,56] as it applies

to atomic-size qubit systems interacting with environment and typically "measured" directly by the effect of the spin-qubit state on transport, or first transferring the spin state to a charge state that is easier to measure, e.g., in single-electron transistors and similar devices.

In this survey, we outline results of the second evaluation task: that of understanding the processes and times scales involved in the dynamics of the actual computation. As summarized in Fig. 3, this main stage of the quantum computation process involves control of spins and their interactions. It also involves processes that we do not control and are trying to minimize: relaxation and decoherence.

Control of individual qubits is usually accomplished externally. For nuclear spins, NMR radio-frequency radiation can be used, see Fig. 2. For electron spins, the ESR microwave frequencies are suitable. Such radiation cannot be focused on the scale of 10–100 nm. Instead, selectivity must be accomplished by independent means. Several proposals exist, the most promising being control by gates. The applied gate voltage modifies the electronic wave function changing interactions and therefore resonant frequencies. We will denote the time scale of the external single-qubit control by $T_{\rm ext}$. This can be the Rabi time of a spin flip.

The qubit–qubit interactions are typically assumed to be mediated by electrons that "visit" both qubit environments. For instance, in liquid-state ensemble NMR [8,9] with complex molecules, or in the original model [32] of phosphorous impurity donors in silicon, the wave functions of the valence, outer electrons of nearby qubits overlap. Specifically, in the P donor case, the single outer electron of the donor atom remains bound at low temperatures but has orbital radius of order 2 nm owing to the large dielectric constant of the silicon host. Therefore, it is hoped that these electrons, in nearby donors positioned as in Fig. 2, will mediate nuclear-spin qubit interactions.

Our approach [27,28] allows for larger qubit separation, up to order 100 nm, by relying on the two-dimensional electron gas in the heterostructure to mediate qubit–qubit interactions. This two-dimensional electron gas is usually obtained by spontaneous or gate-induced transfer of electrons from impurities to the two-dimensional interface layer in which the qubits are positioned. The source impurities are located at some separation from this layer or in the bulk. The two-dimensional electron gas can be made nondissipative in certain ranges of large applied magnetic fields at low temperatures, when these conduction electrons in the layer are in the integer quantum Hall effect state. Owing to this property and also larger qubit separation allowed, we consider this the most promising approach and focus our present review on such systems.

The time scale of the qubit–qubit interactions will be denoted by $T_{\rm int}$. This is the time it takes to accomplish a two-qubit quantum gate, such as CNOT [2–5,57]. Typically for semiconductor quantum computing proposals, $T_{\rm int} < T_{\rm ext}$, and in fact the case with $T_{\rm int} \ll T_{\rm ext}$ has some advantages because one can use several fast single-spin flips to effectively switch interactions of some qubits off over the gate cycle. Another approach to controlling (on/off) of the two-qubit interactions is by gates, see Fig. 2, which affect the two-dimensional electron gas and the localized electron wavefunctions.

However, the same conduction electrons that provide the qubit–qubit interactions, also expose the qubits to the environment, causing relaxation and decoherence. Other interactions will also be present, that play no role in the useful quantum-computing dynamics but contribute to these undesirable processes. Relaxation and decoherence, and their associated time scales, are addressed in the next section.

3. Time scales of relaxation and decoherence

The processes of relaxation and decoherence considered here [21–28] are associated with the dynamics of a small, few-qubit quantum system as it interacts with the environment. Ultimately, for a large, multi-qubit system, many-body quantum chaos-like behavior must also be accounted for, and some advances in model system studies have been reported recently [5, 58]. Our discussion here will be for the few-qubit case mostly because it allows more system-specific investigations for actual quantum-computing proposals.

Dynamical processes that are unwanted in quantum computing, because they result from the environmental influences rather than from the controlled radiation pulses and gate potentials, can proceed on various time scales. In fact, it is not guaranteed that processes of various types, relaxation/thermalization vs. decoher-

ence/dephasing, can even be unambiguously distinctly identified.

At low temperatures, it is generally hoped that thermalization, which requires transfer of energy, slows down. If the fastest such processes proceed on time scales of order T_1, then this time increases at low temperatures because there are less excitations (phonons, electron gas modes, etc.) to couple the small quantum system to the rest of the solid-state host material.

On the other hand, processes that do not require flow of energy to or from the environment, can still effect the phase of the quantum-superposition amplitudes and cause decoherence. These processes can thus proceed faster, on the time scale T_2. While these comments seem to suggest that $T_2 \leqslant T_1$, there is no obvious reason to have generally $T_2 \ll T_1$ at low temperatures.

However, if the spectrum of the dominant excitations mediating the qubit coupling (both to each other and to the host material) has a gap, then we expect that all the relaxation and decoherence processes will be suppressed. Furthermore, the suppression of the relaxation will be exponential, with the Boltzmann factor for that energy gap. Then, $T_2 \ll T_1$ will be satisfied but also, more importantly, the actual values of both time scales will be inordinately large. This was found, theoretically and experimentally, to be the case for the integer-quantum-Hall-state two-dimensional electron gas as mediator of the localized-spin (nuclear, electronic) coupling in semiconductor heterostructures [27,28,59–63].

It is important to emphasize that relaxation and decoherence are really many-body properties of the system plus environment. Entanglement with the environment owing to the unwanted couplings results in the small quantum system having no pure wavefunction even if initially it was prepared in a pure state. Instead, it can be described by a statistical mixture represented by a density matrix, once the environment is traced over.

This reduced density matrix of the system is expected to evolve to the thermal one at large times. The approach to the thermal density matrix, which is diagonal in the system-energy basis, defines the time scale T_1. If the temperature is low enough, then there is the expectation, see [25,26] and references therein, that for some intermediate time scales, of order T_2, the density matrix becomes nearly-diagonal in a basis

Initialize

Control and evolve in time $T_{ext}, T_{int} << T_2, T_1$

- **Control qubits:** T_{ext}
- **& interactions:** T_{int} $(> T_{ext})$
- **Avoid relaxation:** T_1
- **& decoherence:** T_2 $(\leq T_1)$

Measure

Fig. 3. Evaluation of quantum computing models. One of the criteria for feasibility of quantum computing in a given physical system is the possibility of initialization of the qubits in the desired superposition state. Another important design consideration is control of qubit states and of their interactions. In order to implement quantum computing effectively, the time scales for realization of single and two-qubit logic gates, $T_{\rm ext}$ and $T_{\rm int}$, respectively, should be several orders of magnitude smaller than the time scales of relaxation and decoherence, T_1 and T_2. The relationships between these time scales are further explained in the text. Finally, efficient and reliable measurement of the output state of the qubits is required for reading off the result of the computation and presently represents a formidable experimental challenge.

which is determined not by the systems Hamiltonian (energy), but by the interaction operator with the environment. This latter process corresponds to loss of quantum coherence.

As emphasized in Fig. 3, evaluation of a quantum-computing proposal requires, among other things, establishing the relation $T_{\rm ext}, T_{\rm int} \ll T_2, T_1$. Owing to calculational difficulties, the single-qubit times $T_{1,2}$ will usually be used, though, as mentioned earlier, some study of the multi-qubit "quantum chaos" effects may be required. For spin-qubit quantum computing in semiconductor heterostructures, the relation is typically $T_{\rm ext} \ll T_{\rm int} \leqslant T_2 \ll T_1$, so the issue is usually how small is the quality ratio $Q = T_{\rm int}/T_2$.

The required value of Q, needed for fault-tolerant quantum error correction, depends on the physical model of error sources and can be as small as $Q = 10^{-6}$–10^{-4}, see [15,18–20], or as large as $Q = 1/2$, see [64]. For the systems of interest to us here, spin qubits in semiconductor structures, the value of $Q = 10^{-5}$ is a reasonable working estimate. Thus, we seek systems/conditions with $T_{\rm int}/T_2 \leqslant 10^{-5}$.

4. Results for nuclear-spin qubits

In this section we outline results for models of quantum computing with nuclear spins as qubits, and with coupling mediated by the two-dimensional elec-

tron gas in the integer quantum Hall effect state [27, 28,30]. In strong magnetic fields, the spatial states of the electrons confined in the two-dimensional layer in which the qubits are placed, see Fig. 2, are quantized by the field to resemble free-space Landau levels. The lattice potential and the impurities actually cause formation of narrow bands instead of the sharp levels, separated by localized states. As a result, for ranges of magnetic field, the localized states fill up while the extended states resemble completely filled integer number of Landau levels. These states are further Zeeman split owing to the electron spin. At low temperatures, one can find field values such that only one Zeeman sublevel is completely filled in the ground state.

The electronic state in such systems, that show the quantum Hall effect [31] in conductivity, are highly correlated and nondissipative. If nuclear spins are used as qubits, i.e. atoms with nuclear spin $1/2$ are sparsely positioned in the zero-nuclear spin host, such as the zero-nuclear-spin isotope 28 of Si, which constitutes 92% of natural silicone, then their zero-temperature relaxation will be significantly slowed down: experimentally, $T_1 \simeq 10^3$ sec [62].

Localized spins, both nuclear and electronic, interact by exchanges of spin excitons—spin waves consisting of a superposition of bound electron–hole pair states. The spectrum of these excitations [65,66], observed experimentally in [67], has a gap corresponding to the Zeeman splitting. This gap is the cause of slow relaxation and decoherence. The exchange of virtual spin excitons mediates the qubit–qubit interaction and also, via scattering of virtual excitons from impurity potentials, relaxation and decoherence of single qubits.

The original proposal to use nuclear spin qubits directly coupled by the two-dimensional electron gas [30], required positioning the qubits at distances comparable to several magnetic lengths. The latter is of order 10 nm for magnetic fields of several Tesla. The qubit–qubit interaction decays exponentially on this length scale. Recently, we proposed a new improved model [28] in which the qubit interactions are mediated via coupling of the two-dimensional electron gas to the outer impurity electrons. This applies if the atoms, whose nuclear spins are the qubits, are single-electron donors such as the isotope 31 of P. These phosphorous impurities were originally utilized in the model of Kane [32] where they must be actually positioned at separations of about 4 nm for the wavefunctions of the outer electrons, which are bound at low temperatures, to overlap significantly.

In our new improved model [28], with nuclear spins coupling to the outer bound electrons which, in turn, interact via the two-dimensional electron gas, the interaction turned out to be of a much longer range as compared to the model of [32]: the qubit separation can be of order 100 nm. Another advantage is that gate control of the individual qubits and of qubit–qubit interactions is possible. We have carried out extensive perturbative many-body calculations [27,28, 30,68] allowing estimation of $T_{\rm int}$ and T_2 for both the original quantum-computing proposal [30] and its improved version [28], where the main improvement is in the possibility of the gate control along the lines of [32]. The "clock speed" of the improved model is also faster by about two orders of magnitude. The technical details of these rather cumbersome calculations are available in the literature and will not be reviewed here.

The results are summarized in Table 1. We show estimates of all four relevant time scales for the two models introduced earlier. The "original" model [30] corresponds to nuclear spins $1/2$ introduced at qubits in atoms without an outer loosely bound electron. The "improved" model corresponds to the case when the outer electron is present and its interaction with the nuclear spin and the two-dimensional electron gas dominates the dynamics.

The data shown in Table 1 were obtained assuming typical parameters for the standard heterojunctions utilized in quantum-Hall-effect experiments today, and qubit separation of 65 nm. Thus, the parameter values taken [28,30] were more appropriate for the GaAs system than for Si, even though the main isotopes of gallium and arsenic have nuclear spin $3/2$ and cannot serve as spin-zero hosts. The reason

Table 1
Time scales of the qubit dynamics for the original [30] and improved [28] versions of the nuclear spin quantum computer with interactions mediated by the two-dimensional electron gas

	The original model	The improved model
$T_{\rm ext}$	$O(10^{-5})$ sec	$O(10^{-5})$ sec
$T_{\rm int}$	$O(1)$ sec	$O(10^{-2})$ sec
T_1	$O(10^3)$ sec	$O(10)$ sec
T_2	$O(10)$ sec	$O(10^{-1})$ sec

for using these values has been that experimental verification of some of the numbers might be possible in the available materials before cleaner and different composition materials needed for quantum computing are produced.

Our estimates, see Table 1, indicate that the quality factor $Q = 10^{-5}$ is not obtained for the present system. Actually, no quantum computing proposal to date, scalable by other criteria, satisfies the 10^{-5} quality-factor criterion. The values range from 10^{-1} to 10^{-2}. The resolution could come from development of better error-correction algorithms or from improving the physical system to obtain a better quality factor. In our estimation of the decoherence time scale, we used parameters typical of a standard, "dirty" heterostructure with large spatial fluctuations of the impurity potential. These heterostructures have been suitable for standard experiments because they provide wider quantum-Hall plateaus, i.e. ranges of magnetic field for which all the extended states of a Zeeman sublevel are filled. Much cleaner, ultra-high mobility structures can be obtained by placing the ionized impurity layer at a larger distance from the two-dimensional gas or by injecting conduction electrons into the heterostructure by other means. Thus, our quantum-computing proposals [28,30] are unique not only in the large qubit separation allowed but also in that there is a clear direction of exploration to allow physical, rather than algorithmic, resolution of the quality factor problem. This possibility should be further explored both experimentally and theoretically.

Acknowledgements

The authors acknowledge useful discussions and collaboration with M.L. Glasser, R.G. Mani and L.S. Schulman. This research was supported by the National Security Agency (NSA) and Advanced Research and Development Activity (ARDA) under Army Research Office (ARO) contract number DAAD 19-99-1-0342.

References

[1] P.W. Shor, in: Proc. 35th Annual Symposium on the Foundations of Computer Science, IEEE Computer Science Society Press, Los Alamitos, CA, 1994, p. 124.
[2] D.P. DiVincenzo, Science 270 (1995) 255.
[3] A. Ekert, R. Jozsa, Rev. Mod. Phys. 68 (1996) 733.
[4] A.M. Steane, Rept. Prog. Phys. 61 (1998) 117.
[5] G.P. Berman, G.D. Doolen, R. Mainieri, V.I. Tsifrinovich, Introduction to Quantum Computers, World Scientific, Singapore, 1998.
[6] S. Datta, Electronic Transport in Mesoscopic Systems, Cambridge University Press, 1997.
[7] L.K. Grover, Phys. Rev. A 79 (1997) 4709.
[8] N.A. Gershenfeld, I.L. Chuang, Science 275 (1997) 350.
[9] D.G. Cory, A.F. Fahmy, T.F. Havel, Proc. Natl. Acad. Sci. USA 94 (1997) 1634.
[10] A. Barenco, Proc. Roy. Soc. London A 449 (1995) 679.
[11] A. Barenco, C.H. Bennett, R. Cleve, D.P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J.A. Smolin, H. Weinfurter, Phys. Rev. A 52 (1995) 3457.
[12] D.P. DiVincenzo, Phys. Rev. A 51 (1995) 1015.
[13] S. Lloyd, Phys. Rev. Lett. 75 (1995) 346.
[14] P.W. Shor, in: Proc. 37th Annual Symposium on the Foundations of Computer Science, IEEE Computer Science Society Press, Los Alamitos, CA, 1996, p. 56.
[15] D. Aharonov, M. Ben-Or, Fault-tolerant quantum computation with constant error, Preprint, available at http://arxiv.org/abs/quant-ph/9611025, extended version at http://arxiv.org/abs/quant-ph/9906129.
[16] A. Steane, Phys. Rev. Lett. 78 (1997) 2252.
[17] D.P. DiVincenzo, in: L. Kowenhoven, G. Schoen, L. Sohn (Eds.), Mesoscopic Electron Transport, NATO Adv. Study Inst. Series E, Kluwer, Dordrecht, 1997.
[18] E. Knill, R. Laflamme, Phys. Rev. A 55 (1997) 900.
[19] D. Gottesman, Phys. Rev. A 57 (1998) 127.
[20] J. Preskill, Proc. Roy. Soc. London A 454 (1998) 385.
[21] A. Garg, Phys. Rev. Lett. 77 (1996) 764.
[22] G.M. Palma, K.A. Suominen, A.K. Ekert, Proc. Roy. Soc. London A 452 (1996) 567.
[23] I.S. Tupitsyn, N.V. Prokof'ev, P.C.E. Stamp, Int. J. Modern Phys. B 11 (1997) 2901.
[24] T. Maniv, Yu.A. Bychkov, I.D. Vagner, P. Wyder, Fast incomplete decoherence of nuclear spins in quantum hall ferromagnet, Preprint, available at http://arxiv.org/abs/cond-mat/0011361.
[25] D. Mozyrsky, V. Privman, J. Stat. Phys. 91 (1998) 787.
[26] D. Mozyrsky, V. Privman, Modern Phys. Lett. B 14 (2000) 303.
[27] D. Mozyrsky, V. Privman, I.D. Vagner, Phys. Rev. B 63 (2001) 085313.
[28] D. Mozyrsky, V. Privman, M.L. Glasser, Indirect interaction of solid-state qubits via two-dimensional electron gas, Preprint, available at http://arxiv.org/abs/cond-mat/0012470.
[29] D. Loss, D.P. DiVincenzo, Phys. Rev. A 57 (1998) 120.
[30] V. Privman, I.D. Vagner, G. Kventsel, Phys. Lett. A 239 (1998) 141.
[31] R.E. Prange, S.M. Girvin (Eds.), The Quantum Hall Effect, Springer-Verlag, New York, 1987.
[32] B.E. Kane, Nature 393 (1998) 133.
[33] A. Imamoglu, D.D. Awschalom, G. Burkard, D.P. DiVincenzo, D. Loss, M. Sherwin, A. Small, Phys. Rev. Lett. 83 (1999) 4204.

[34] R. Vrijen, E. Yablonovitch, K. Wang, H.W. Jiang, A. Balandin, V. Roychowdhury, T. Mor, D.P. DiVincenzo, Phys. Rev. A 62 (2000) 012306.
[35] G.P. Berman, G.D. Doolen, P.C. Hammel, V.I. Tsifrinovich, Solid-state nuclear spin quantum computer based on magnetic resonance force microscopy, Preprint, available at http://arxiv.org/abs/quant-ph/9909033.
[36] S. Bandyopadhyay, Phys. Rev. B 61 (2000) 13 813.
[37] J.H. Reina, L. Quiroga, N.F. Johnson, Quantum information processing in semiconductor nanostructures, Preprint, available at http://arxiv.org/abs/quant-ph/0009035.
[38] R. Ionicioiu, G. Amaratunga, F. Udrea, Quantum computation with ballistic electrons, Preprint, available at http://arxiv.org/abs/quant-ph/0011051.
[39] J. Levy, Quantum information processing with ferroelectrically coupled quantum dots, Preprint, available at http://arxiv.org/abs/quant-ph/0101026.
[40] X. Hu, S. Das Sarma, Solid state quantum computation with spin qubits: Can it be done in multielectron dots?, Preprint, available at http://arxiv.org/abs/cond-mat/0101102.
[41] X. Hu, S. Das Sarma, Theoretical issues in spin-based quantum dot quantum computation, Preprint, available at http://arxiv.org/abs/cond-mat/0102019.
[42] Yu. Makhlin, G. Schoen, A. Shnirman, Nature 398 (1999) 305.
[43] J. Mooij, T. Orlando, L. Levitov, L. Tian, V. van der Wal, S. Lloyd, Science 285 (1999) 1036.
[44] Y. Nakamura, Yu.A. Pashkin, J.S. Tsai, Nature 398 (1999) 786.
[45] J.R. Friedman, V. Patel, W. Chen, S.K. Tolpygo, J.E. Lukens, Detection of a Schröedinger's cat state in an rf-SQUID, Preprint, available at http://arxiv.org/abs/cond-mat/0004293.
[46] D.V. Averin, Quantum computing and quantum measurement with mesoscopic Josephson junctions, Preprint, available at http://arxiv.org/abs/quant-ph/0008114.
[47] P.M. Platzman, M.I. Dykman, Science 284 (1999) 1967.
[48] J.A. Jones, M. Mosca, J. Chem. Phys. 109 (1998) 1648.
[49] J.R. Tucker, T.-C. Shen, Solid State Electron. 42 (1998) 1061.
[50] B.E. Kane, N.S. McAlpine, A.S. Dzurak, R.G. Clark, G.J. Milburn, H.B. Sun, H. Wiseman, Phys. Rev. B 61 (2000) 2961.
[51] R.G. Clark, unpublished results presented at a quantum-computing conference held in Baltimore, MD, August 2000.
[52] R. Vrijen, E. Yablonovitch, A spin-coherent semiconductor photo-detector for quantum communication, Preprint, available at http://arxiv.org/abs/quant-ph/0004078.
[53] H.W. Jiang, E. Yablonovitch, Gate-controlled electron spin resonance in a GaAs/AlGaAs heterostructure, Preprint, available at http://arxiv.org/abs/quant-ph/0102044.
[54] H. Kosaka, A.A. Kiselev, F.A. Baron, K.W. Kim, E. Yablonovitch, Electron g-factor engineering in III–V semiconductors for quantum communications, Preprint, available at http://arxiv.org/abs/quant-ph/0102056.
[55] A.N. Korotkov, D.V. Averin, Continuous weak measurement of quantum coherent oscillations, Preprint, available at http://arxiv.org/abs/cond-mat/0002203.
[56] A.N. Korotkov, Selective quantum evolution of a qubit state due to continuous measurement, Preprint, available at http://arxiv.org/abs/cond-mat/0008461.
[57] V. Privman, D. Mozyrsky, S.P. Hotaling, in: Proc. Conf. Photonic Quantum Computing. AeroSense 97, SPIE Proc., Vol. 3076, SPIE—The International Society for Optical Engineering, 1997, p. 84.
[58] G.P. Berman, F. Borgonovi, F.M. Izrailev, V.I. Tsifrinovich, Avoiding quantum chaos in quantum computation, Preprint, available at http://arxiv.org/abs/quant-ph/0012106.
[59] I.D. Vagner, T. Maniv, Phys. Rev. Lett. 61 (1988) 1400.
[60] R. Tycko, S.E. Barrett, G. Dabbagh, L.N. Pfeiffer, K.W. West, Science 268 (1995) 1460.
[61] S.E. Barrett, G. Dabbagh, L.N. Pfeiffer, K.W. West, R. Tycko, Phys. Rev. Lett. 74 (1995) 5112.
[62] A. Berg, M. Dobers, R.R. Gerhards, K.V. Klitzing, Phys. Rev. Lett. 64 (1990) 2563.
[63] S.V. Iordanskii, S.V. Meshkov, I.D. Vagner, Phys. Rev. B 44 (1991) 6554.
[64] E. Knill, R. Laflamme, G. Milburn, Thresholds for linear optics quantum computation, Preprint, available at http://arxiv.org/abs/quant-ph/0006120.
[65] Yu.A. Bychkov, S.V. Iordanskii, G.M. Eliashberg, JETP Lett. 33 (1981) 143.
[66] C. Kallin, B.I. Halperin, Phys. Rev. B 30 (1984) 5655.
[67] A. Pinczuk, B.S. Dennis, D. Heiman, C. Kallin, L. Brey, C. Tejedor, S. Schmitt-Rink, L.N. Pfeiffer, K.W. West, Phys. Rev. Lett. 68 (1992) 3623.
[68] Yu.A. Bychkov, T. Maniv, I.D. Vagner, Solid State Commun. 94 (1995) 61.

ELSEVIER

Computer Physics Communications 146 (2002) 339–344

Computer Physics Communications

www.elsevier.com/locate/cpc

Towards a NMR implementation of a quantum lattice gas algorithm

Marco A. Pravia [a], Zhiying Chen [a], Jeffrey Yepez [b], David G. Cory [a,*]

[a] *Department of Nuclear Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*
[b] *Air Force Research Laboratory, Hanscom Field, MA 01731, USA*

Received 20 August 2001

Abstract

Recent theoretical results suggest that an array of quantum information processors communicating via classical channels can be used to solve fluid dynamics problems. Quantum lattice-gas algorithms (QLGA) running on such architectures have been shown to solve the diffusion equation and the nonlinear Burgers equations. In this report, we describe progress towards an ensemble nuclear magnetic resonance (NMR) implementation of a QLGA that solves the diffusion equation. The methods rely on NMR techniques to encode an initial mass density into an ensemble of two-qubit quantum information processors. Using standard pulse techniques, the mass density can then manipulated and evolved through the steps of the algorithm. We provide the experimental results of our first attempt to realize the NMR implementation. The results qualitatively follow the ideal simulation, but the observed implementation errors highlight the need for improved control.

1. Introduction

The field of quantum information processing (QIP) has made steady progress over the past decade, driven in part by the realization that some quantum algorithms offer a computational advantage over the best known classical counterparts [1]. To reach a practical improvement, however, most quantum algorithms require a large number of qubits coupled quantum mechanically, making physical implementation difficult. Recently, however, it has been suggested that some interesting problems might be solvable by a hybrid classical-quantum device defined as a type-II quantum computer [2]. A type-II quantum computer is essentially a parallel lattice of small quantum information processors that share information through classical channels. Such a device offers the experimental simplification that quantum coherences need only be maintained locally within each small quantum processor. Using this architecture, it might be possible to increase the range of problems that small quantum processors can tackle by classically stringing many of them together. A type-II quantum computer may thus serve as an intermediate architecture between few-qubit and large-scale quantum computers.

In this report, we explore the experimental aspects of building a type-II quantum computer using nuclear magnetic resonance (NMR) techniques. Quantum information processing by NMR usually employs a liquid sample of molecules containing spin-$\frac{1}{2}$ nuclei that is subjected to a strong magnetic field [3]. A typi-

* Corresponding author.
E-mail address: dcory@mit.edu (D.G. Cory).

0010-4655/02/$ – see front matter
PII: S0010-4655(02)00425-3

cal field B_0 of $\sim$10 T creates an energy difference ΔE between the aligned and anti-aligned spin states that drives the system to an equilibrium state with net magnetization. At room temperature, $\Delta E/kT$ is about 10^{-5}, so that the net magnetization is relatively small, but, given the large number of molecules in the sample ($\sim 10^{18}$), it is still easily detectable. The entire spin ensemble is accurately described by a reduced density matrix of only the intramolecular spin degrees of freedom. The ensemble nature of the NMR sample thus makes it inherently applicable to parallel computation. A type-II architecture can be mapped onto an NMR sample by creating a correspondence between the sites of the lattice and spatially distinct spin ensembles. Using magnetic field gradients and radiofrequency (RF) pulses, information in the lattice can be encoded, manipulated, and read out. As a first test of the NMR implementation, we chose a basic quantum lattice gas algorithm (QLGA) that solves diffusive dynamics in one dimension.

2. Lattice-gas system

The diffusion of a mass density ρ is governed by

$$\frac{\partial \rho}{\partial t} = D\frac{\partial^2 \rho}{\partial t^2}. \tag{1}$$

The above equation corresponds to the macroscopic effective field theory result. Its relation to the lattice-gas dynamics is seen by breaking space into an array of lattice sites with occupation probabilities assigned to each site [4,5]. The ensemble average mesoscopic dynamics are controlled by the transport equations [2]

$$f_1(n, m+1) = f_1(n, m) + \tfrac{1}{2}\big[f_2(n, m) - f_1(n, m)\big], \tag{2}$$

$$f_2(n, m+1) = f_2(n, m) - \tfrac{1}{2}\big[f_2(n, m) - f_1(n, m)\big], \tag{3}$$

where f_1 and f_2 represent occupation probabilities and the bracketed terms represent a collision operator. The number density ρ is the sum of f_1 and f_2. The indices n and m correspond to lattice site and time step, respectively. The connection between the diffusion equation and the transport equations may be seen by taking the Chapman–Enskog expansion of the lattice Boltzmann equation written in terms of occupation probabilities.

3. Quantum lattice-gas algorithm

The quantum lattice-gas implementation relies on mapping each initial occupation probability f_1 and f_2 into the corresponding single-particle states of two quantum bits,

$$\big|q_{1,2}(n, m)\big\rangle = \sqrt{f_{1,2}(n, m)}\,|1\rangle + \sqrt{1 - f_{1,2}(n, m)}\,|0\rangle, \tag{4}$$

where $|q_{1,2}(n, m)\rangle$ are the qubit states and $|0\rangle$ and $|1\rangle$ correspond to the eigenstates of a two-level system. The resulting two-qubit wave function for a site becomes

$$\begin{aligned}\big|\psi(n, m)\big\rangle = {} & \sqrt{f_1 f_2}\,|11\rangle + \sqrt{f_1(1 - f_2)}\,|10\rangle \\ & + \sqrt{(1 - f_1) f_2}\,|01\rangle + \sqrt{(1 - f_1)(1 - f_2)}\,|00\rangle\end{aligned} \tag{5}$$

where $|\psi(n, m)\rangle$ spans the Hilbert space of two coupled quantum systems. After initialization, the algorithm calls for a collision operation

$$\big|\psi'(n, m)\big\rangle = \widehat{U}\big|\psi(n, m)\big\rangle \tag{6}$$

that is carried out via unitary evolution by a "square-root of swap" gate $\widehat{U}$. The gate $\widehat{U}$ can be written as

$$\widehat{U} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} - \frac{\mathrm{i}}{2} & \frac{1}{2} + \frac{\mathrm{i}}{2} & 0 \\ 0 & \frac{1}{2} + \frac{\mathrm{i}}{2} & \frac{1}{2} - \frac{\mathrm{i}}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{7}$$

in the standard basis. The next step in the computation requires a measurement of the occupation numbers

$$f'_{1,2}(n, m) = \big\langle \psi'(n, m)\big|\hat{n}_{1,2}\big|\psi'(n, m)\big\rangle, \tag{8}$$

where the number operators $\hat{n}_{1,2}$ are defined as

$$\hat{n}_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \hat{n}_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{9}$$

The measured occupation numbers f'_1 and f'_2 are then streamed to the nearest lattice sites in opposite directions, as given by

$$f_1(n, m+1) = f'_1(n+1, m), \tag{10}$$

$$f_2(n, m+1) = f'_2(n-1, m). \tag{11}$$

The entire diffusion algorithm can be summarized in four steps:

(1) Initialization of occupation probabilities in each spatially distinct site.
(2) Application of the collision operator, $\widehat{U}$, at all sites.
(3) Readout of the expectation values of the number operators.
(4) Determination of the new occupation probabilities by streaming to nearest neighbors.

4. NMR implementation

In our particular test, we implemented a two-qubit diffusion algorithm using a solution of chloroform ($^{13}CHCl_3$) where the hydrogen and the labeled carbon nuclei served as qubits 1 and 2, respectively. Fig. 1 shows the energy level diagram of the spins and a picture of the molecule. As shown in the diagram, the proton splitting is four times larger than the carbon splitting, and both splittings are a small fraction of kT.

4.1. Mapping to spin ensembles

The first step in creating an experiment to study the implementation of type-II quantum computer is to define a mapping of the theoretically required quantum states to a real physical system. In the liquid-state NMR case, the required quantum states $|\psi(n,m)\rangle$ are physically encoded onto spin ensembles described by density matrices $\sigma(n,m)$

$$|\psi(n,m)\rangle \to \sigma(n,m). \tag{12}$$

However, the thermal equilibrium of liquid-state NMR systems is a highly mixed state that is not immediately applicable to quantum computing experiments. As a result, the thermal equilibrium state must first be reset to a pseudopure state of the form [6,7]

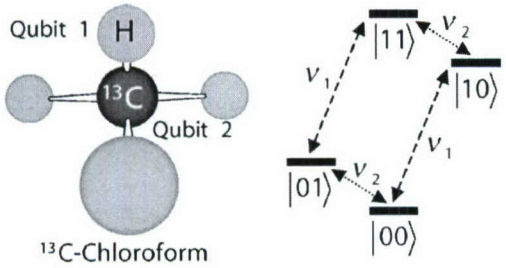

Fig. 1. The picture show the chloroform molecule and the nuclear spin energy level diagram.

$$\sigma(n,m) = 1 - \varepsilon\big|\psi(n,m)\big\rangle\big\langle\psi(n,m)\big|. \tag{13}$$

The above pseudopure state transforms identically to the corresponding pure state $|\psi(n,m)\rangle$. Each subensemble $\sigma(n,m)$ is in turn composed of a large number ($\sim 10^{18}$) of individual molecules distributed within a slice of a cylindrical sample. More formally, the reduced density matrix $\sigma(n,m)$ at a site is

$$\sigma(n,m) = \mathrm{Tr}_{\vec{r}}\Big[\sum_{\vec{r}} T\big[z - \Delta z\big(n - \tfrac{1}{2}\big)\big] \times \big|\phi(\vec{r},m)\big\rangle\big\langle\phi(\vec{r},m)\big|\Big], \tag{14}$$

where $|\phi(\vec{r},m)\rangle$ is the nuclear spin state of a single molecule located at position $\vec{r}$, $T(z)$ is the "top hat" function

$$T(z) = \begin{cases} 1, & |z| \leqslant \frac{1}{2}, \\ 0, & |z| > \frac{1}{2} \end{cases} \tag{15}$$

that selects the relevant spatial slice with thickness Δz, and $\mathrm{Tr}_{\vec{r}}$ denotes the partial trace over the spatial degree of freedom. The variable z represents the corresponding coordinate of the vector $\vec{r}$. The states $\sigma(n,m)$ required at the beginning of each update are created by applying shaped radiofrequency (RF) pulses in the presence of linear magnetic field gradients. This step is related to slice-selection in magnetic resonance imaging (MRI). Fig. 2 depicts the geometrical arrangement of the slices relative to the gradient and RF coils in the NMR probe.

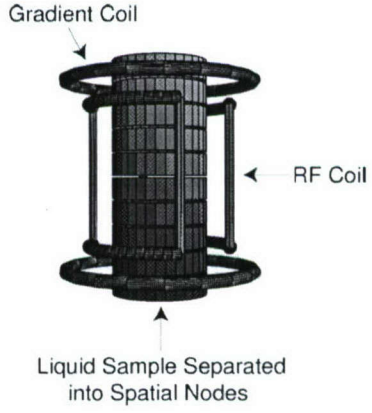

Fig. 2. The cylindrical sample of chloroform employed in this experiment is addressed in slices by the combined action of magnetic field gradients and shaped RF pulses. Each slice represents a node in the lattice of quantum information processors.

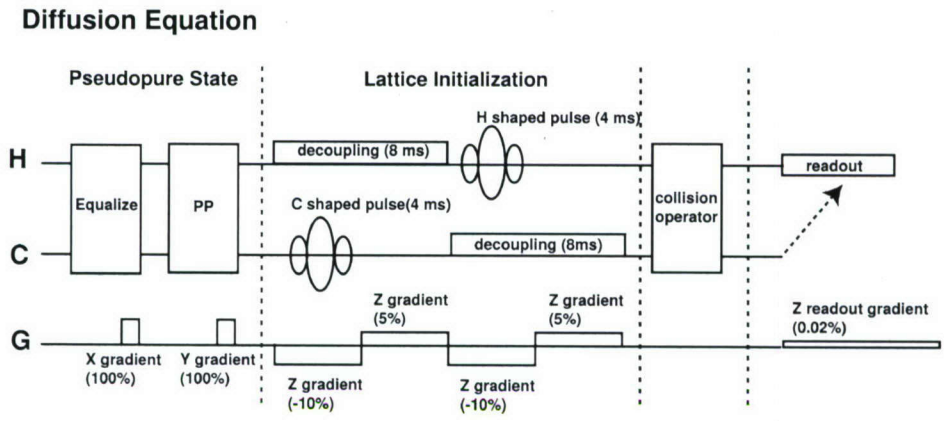

Fig. 3. The pulse sequence for a single time step of the algorithm begins with the pseudopure state preparation. Gradients are used to perform the necessary non-unitary operations required for equalizing the magnetization of the two-spin species and to prepare the pseudo pure state. The lattice initialization is accomplished by applying weak RF shapes in the presence of a magnetic field gradient in the Z direction. A decoupling sequence prevents the scalar coupling from interfering with the initialization. The collision operation is performed by a sequence of coupling delay and strong RF pulses. The collision pulse sequence is applied without a gradient so that all of the spins in the lattice feel the same operation. Readouts of both the carbon and hydrogen magnetizations are carried out on the hydrogen channel in two separate experiments.

4.2. Control and measurement of spin system

In the absence of a magnetic field gradient, the Hamiltonian of the spin system in the doubly-rotating frame is

$$H(t) = \frac{\pi J}{2}\sigma_z^1\sigma_z^2 + \left[w_x^1(t)\sigma_x^1 + w_y^1(t)\sigma_y^1\right] + \left[w_x^2(t)\sigma_x^2 + w_y^2(t)\sigma_y^2\right]. \quad (16)$$

The first term denotes the scalar interaction between the spins, while the remaining terms are the externally-controlled RF Hamiltonian. The operators of the form $\sigma_{x,y,z}^{1,2}$ are Pauli spin matrices corresponding to each qubit, and the scalar coupling Hamiltonian is a Kronecker product of the single-spin operators. The RF part of the Hamiltonian generates arbitrary single spin rotations with high fidelity when the nutation rates $w_{x,y}^{1,2}$ are much stronger than J, the scalar coupling constant.

As mentioned before, the collision operator $\widehat{U}$ for the diffusion algorithm is the square-root of swap gate. The unitary operator $\widehat{U}$ can be written as

$$\widehat{U} = \exp\left[-\mathrm{i}\frac{\pi}{8}\left(\sigma_x^1\sigma_x^2 + \sigma_y^1\sigma_y^2 + \sigma_z^1\sigma_z^2\right)\right] \quad (17)$$

if an irrelevant phase is ignored. Written in this form, it is clear that $\widehat{U}$ can be decomposed into the product of three commuting terms. Each term can be implemented by making use of the scalar coupling Hamiltonian plus appropriate single-spin rotations [8]. The operator $\widehat{U}$ is applied to all the sub-ensembles $\sigma(n,m)$ such that

$$\sigma'(n,m) = \widehat{U}\sigma(n,m)\widehat{U}^\dagger. \quad (18)$$

The final steps of the algorithm are to read the occupation numbers encoded in $\sigma'(n,m)$ and to stream them to nearby sites. The readout is accomplished by noticing that Eq. (8) can be rewritten in terms of the z-Pauli matrices as

$$f'_{1,2}(n,m) = \left\langle \psi'(n,m)\left|\frac{1-\sigma_z^{1,2}}{2}\right|\psi'(n,m)\right\rangle = \tfrac{1}{2}\left[1 - \left\langle\psi'(n,m)\left|\sigma_z^{1,2}\right|\psi'(n,m)\right\rangle\right] \quad (19)$$

using the fact that $\hat{n}_{1,2} = \frac{1}{2}(1-\sigma_z^{1,2})$. The last equation can be written in the final form

$$f'_{1,2}(n,m) = \tfrac{1}{2}\left[1 - M_z^{1,2}\right], \quad (20)$$

where the trace has been replaced by the z-magnetization $M_z^{1,2}$. The z-magnetization is measured in NMR by applying a "read" $\pi/2$ pulse and observing the transverse magnetization. The measured values $f'_{1,2}$ can be streamed on a classical computer and then reinitialized onto the lattice.

4.3. Pulse sequence

The diagram in Fig. 3 shows the main parts of a single time step of the NMR implementation: pseudopure

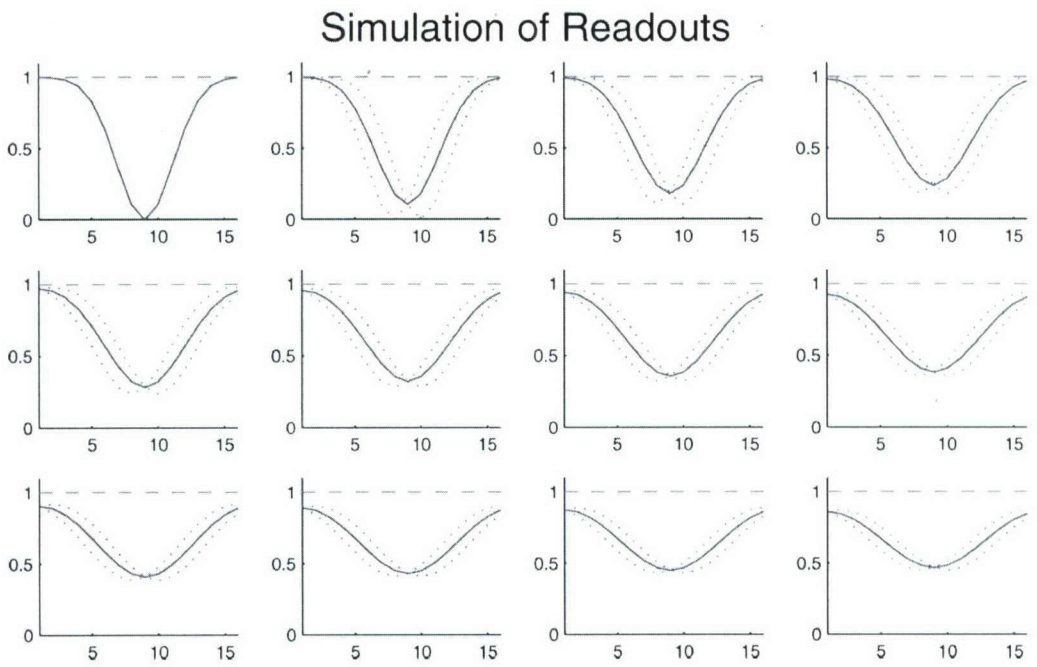

Fig. 4. The plots show a simulated run of the ideal quantum lattice gas algorithm for diffusion. The top left plot contains the first step, followed to the right and then down the rows by subsequent time steps. The dashed lines represent the initialized occupation numbers $f_{1,2}$ for each spin, while the solid lines represent the occupation numbers $f'_{1,2}$ present after the collision. The x-axis labels the node number and runs from 1 to 16.

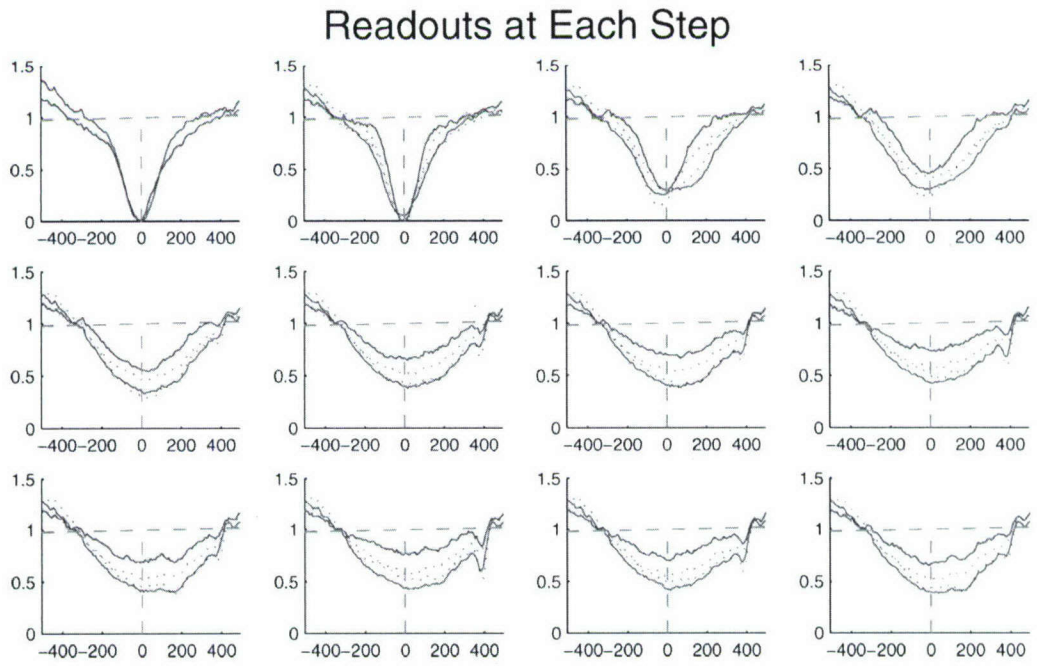

Fig. 5. The experimental results for the corresponding time steps of the simulations from Fig. 4. Although the general features follow the simulation, the experimental results are not of high fidelity and suggest a need for more precise control. The x-axis labels the observed spectral frequency. The actual nodes used in the experiment reside in the region between -200 and 200 Hz. The outlying region is included for reference.

state preparation, lattice initialization, collision, and readout. The top two lines correspond to operations on the two qubits (H and C), while the third line shows the required gradient pulses. The pseudopure state was prepared by first equalizing the magnetizations of the two spins, followed by a pseudopure state creation sequence [9]. The starting occupation numbers for each time step were then encoded using weak shaped RF pulses on the two spins. Because the RF power utilized was weak relative to the gradient strength, the shape of the pulse was determined by taking the Fourier transform of the desired magnetization profile [10]. A decoupling sequence was applied simultaneously with the RF shape, to average out the effects of the scalar coupling on the RF excitation.

The collision operator was implemented by decomposing the total unitary operator into sequences of scalar coupling delays and RF pulses. The readout was performed by recording the spectra in the presence of a weak gradient. The classical communication part of the algorithm was absorbed into the encoding operation of the next time step. A linear phase ramp was added to the RF shape, effectively shifting the frequency of the excitation. Since the data on the two spins was to be shifted in different directions, the phase ramps for the two RF pulses had opposite slopes.

5. Results

The two Figs. 4 and 5 show the results of preliminary experiments and, for comparison, the corresponding ideal simulation of the NMR implementation. The experiments where performed using 16 nodes iterated through 12 time steps of the algorithm. As can be seen, the broad features of the diffusion can be seen, but large errors are present in the implementation. The errors are caused by problems in the decoupling sequence, errors in the Fourier transform approximation, and other experimental imperfections. We are continuing to refine the experiments and we expect to correct these errors in the near future with improved results to be published in a subsequent paper [11].

6. Conclusion

Ensemble NMR techniques have been successfully used to study the experimental details involved in quantum information processing. The astronomical number of individual quantum systems ($\sim 10^{18}$) present in typical liquid-state spin ensembles greatly facilitates the problem of measuring spin quantum coherences. In addition, the ensemble nature of the system has been successfully utilized to create the necessary pseudo-pure states and to systematically generate non-unitary operations over the ensemble [12]. In this implementation, we again exploit the ensemble nature, but this time as a means of realizing a lattice of quantum information processors. The implementation combines the advantages of quantum computation at each node with parallel computation throughout the lattice. The large size of the NMR ensemble provides, in principle, sufficient room to explore large lattices. Although achieved experimental results point to the need for better control, the experiments are a first step towards realizing the quantum lattice gas algorithm on a NMR quantum information processor.

References

[1] C.H. Bennett, D.P. DiVincenzo, Nature 404 (2000) 247–255.
[2] J. Yepez, Internat. J. Modern Phys. C 12 (9) (2001) 1–12.
[3] R.R. Ernst, G. Bodenhausen, A. Wokaun, Principles of Nuclear Magnetic Resonance in One and Two Dimensions, Oxford University Press, Oxford, 1994.
[4] J. Yepez, Internat. J. Modern Phys. C 9 (8) (1998) 1587–1596.
[5] J. Yepez, Lecture Notes in Comput. Sci. 1509 (1999) 35–60.
[6] D.G. Cory, A.F. Fahmy, T.F. Havel, Proc. Nat. Acad. Sci. 94 (5) (1997) 1634–1639.
[7] N.A. Gershenfeld, I.L. Chuang, Science 275 (1997) 350–356.
[8] D.G. Cory, M.D. Price, T.F. Havel, Physica D 120 (1–2) (1998) 82–101.
[9] M.A. Pravia, E. Fortunato, Y. Weinstein, M.D. Price, G. Teklemariam, R.J. Nelson, Y. Sharf, S. Somaroo, C.H. Tseng, T.F. Havel, D.G. Cory, Concepts Magnetic Res. 11 (4) (1999) 225–238.
[10] A. Sodickson, D.G. Cory, Progress Magnetic Res. 33 (1998) 78–108.
[11] M.A. Pravia, Z. Chen, J. Yepez, D.G. Cory, In preparation, 2002.
[12] T.F. Havel, Y. Sharf, L. Viola-L, D.G. Cory, Phys. Lett. A 280 (5–6) (2001) 282–288.